CYBER SECURITY

For beginners

CRYPTOGRAPHY FUNDAMENTALS & NETWORK SECURITY

HUGO HOFFMAN

VISIT…

CYBERSECURITY FOR BEGINNERS

CRYPTOGRAPHY FUNDAMENTALS
&
NETWORK SECURITY

BY

HUGO HOFFMAN

All rights reserved.

All rights reserved.

No part of this book may be reproduced in any form or by any electronic, print or mechanical means, including information storage and retrieval systems, without permission in writing from the publisher.

Copyright © 2020

Disclaimer

Professionals should be consulted as needed before undertaking any of the action endorsed herein. Under no circumstances will any legal responsibility or blame be held against the publisher for any reparation, damages, or monetary loss due to the information herein, either directly or indirectly. This declaration is deemed fair and valid by both the American Bar Association and the Committee of Publishers Association and is legally binding throughout the United States. There are no scenarios in which the publisher or the original author of this work can be in any fashion deemed liable for any hardship or damages that may befall the reader or anyone else after undertaking information described herein. The information in the following pages is intended only for informational purposes and should thus be thought of as universal. As befitting its nature, it is presented without assurance regarding its continued validity or interim quality. Trademarks that are mentioned are done without written consent and can in no way be considered an endorsement from the trademark holder.

Intended Audience

This book is designed to anyone who wishes to become an IT Professional, specifically in the field of Information Security. This book is written in everyday English, and no technical background is necessary. If you are a beginner to Informational Technology or Information Security, the contents in this book will provide a high level overview of network and wireless security. If you are preparing to become an IT Professional, such as an Ethical Hacker, IT Security Analyst, IT Security Engineer, Network Analyst, Network Engineer, or a Cybersecurity Specialist, yet still in doubt and want to know about network security, you will find this book extremely useful. You will learn key concepts and methodologies revolving around network Security, as well as key Technologies you should be mindful. If you are truly interested in becoming an Cybersecurity Specialist, this book is for you. Assuming you are preparing to become an Information Security Professional, this book will certainly provide great details that will benefit you as you enter this industry.

Introduction

This book is for anyone who has decided to become an IT Security Professional. This could be anyone who works in IT security, too, as it relates to data networking. If you work in network security in some other aspect of the business, but you're not familiar with Cisco technology, then this is a right starting place if you'd like to learn more about Cisco network security. If you are a security candidate, network security professional, or you have any similar title, you're in the right place. As a prerequisite, I would recommend you to have some basic networking fundamentals or being an entry network technician. The least is to have some necessary computer networking skills, so you understand IP addresses, and capable of identifying networking devices. If you already have an in-depth networking knowledge, that’s great. You also should have some basic understanding of computer systems, and you need to have some necessary internet navigation skills that'll help you out as you go through this book. First, we're going to look at some universal security principles such as confidentiality, integrity, and availability and discuss what those terms mean, how they work together, and why we must understand them. We're also going to talk about SIEM or Security Incident Event Management and Monitoring. We're also going to cover some general security terms that it's going to be vital for you to understand, then talk about standard security zones that we see in the data networking world, and discuss how they relate to specific topologies within the infrastructure.

As we move on, we are going to talk about various threats to our networks, both that we see today as well as some threats that we have seen in the past and discuss why they're not an issue anymore. We're also going to talk about some data loss vectors and how we can mitigate that, but we're also going to look at tools that are used in hacking. Next, we're going to cover hashing and the process of encryption. We'll also take a look at how encryption works and discuss different encryption algorithms, and how keys are used in symmetric and asymmetric algorithms. Then, we will discuss digital signatures and the PKI infrastructure. After that, we are going to look at different kinds of VPN technologies and how we can secure endpoints and remote clients. Next, we are going to look at Firewalls, Intrusion Prevention Systems, Intrusion Detection Systems, various Proxies, and content-based security. We are also going to cover Privilege levels, Syslogs, Reporting, Monitoring, Netflow, and SNMP traffic. After that, we will cover various Authenticating Protocols,

such as Mac Address Bypassing, 802.1X Authentication, and BYOD Security. We were also going to look at multiple technologies to provide availability, such as High Availability, Failover, and Clustering. Lastly, we are going to cover the Security Intelligence basics, Blacklisting, Whitelisting, Email Security, and Data Loss Prevention. If you are ready, let's start looking at confidentiality.

Table of Contents

Chapter 1 Confidentiality

Chapter 2 Integrity

Chapter 3 Availability

Chapter 4 Security Incident Events and Monitoring

Chapter 5 Security Terminologies

Chapter 6 Security Zones

Chapter 7 TCP SYN Flood attack

Chapter 8 Ping of death attack

Chapter 9 Botnet

Chapter 10 IP & MAC Address Spoofing

Chapter 11 DHCP Server & Client Spoofing

Chapter 12 Social Engineering & Phishing

Chapter 13 Spear phishing, Whaling & Pharming

Chapter 14 Watering hole attack & Smishing

Chapter 15 Brute Force & Dictionary Attacks

Chapter 16 Recon Attacks

Chapter 17 Buffer Overflow Attack

Chapter 18 Man-in-the-middle attacks

Chapter 19 Malware & Trojans

Chapter 20 Data Loss Vectors

Chapter 21 Well Known Hacking Tools

Chapter 22 Cryptography Basics

Chapter 23 Substitution Ciphers

Chapter 24 Viginere and Hebern

Chapter 25 Enigma Cypher

Chapter 26 Hashing & MD5 Checksum

Chapter 27 Authenticating Crypto using Hashes

Chapter 28 Pros & Cons of Hash Algorithms

Chapter 29 Encryption Basics

Chapter 30 Breaking Cipher Text

Chapter 31 Encryption and Keys

Chapter 32 Digital Signatures

Chapter 33 Network Topologies & Firewalls

Chapter 34 Intrusion Prevention System

Chapter 35 Content Security Basics

Chapter 36 Remote Access VPN
Chapter 37 DMVPN & Site-to-site VPN
Chapter 38 Securing Endpoints
Chapter 39 Managing Network Devices
Chapter 40 AAA
Chapter 41 ACS & ISE
Chapter 42 Privilege Levels
Chapter 43 Syslog & Reporting
Chapter 44 CPU Threshold, Netflow & SNMP
Chapter 45 Control Plane Policing
Chapter 46 Authenticating Routing Protocols
Chapter 47 802.1X Authentication
Chapter 48 BYOD Security
Chapter 49 Introduction to Firewalls
Chapter 50 Stateless Firewalls
Chapter 51 Stateful Firewalls
Chapter 52 Proxy Servers
Chapter 53 Next Generation Firewalls
Chapter 54 High Availability & Failover
Chapter 55 Clustering
Chapter 56 Zone-based Firewalls
Chapter 57 IDS & IPS
Chapter 58 Security Intelligence Blacklisting
Chapter 59 Email Security
Chapter 60 Data Loss Prevention
Conclusion
About the Author

Chapter 1 Confidentiality

When we talk about confidentiality, integrity, and availability, the three of these together, we'll use the term CIA. CIA is how you might hear that term from various security blueprints is referred to. To describe confidentiality, integrity, and availability, let's begin talking about confidentiality.

Many times the term confidentiality we hear is related to encryption, and when we talk about encryption, we're talking about the ability to hide or privatize our data. Sometimes we'll use the term VPN or virtual private network, and the idea is to keep things private. In terms of encryption, there are several algorithms that we can use.

When we get into cryptography concepts, we're going to talk about algorithms and such, but what we might be benefiting from at this point is talking about them a little bit ahead of time. We can use DES or Data Encryption Standard, which is 56-bit encryption.

We also have 3DES, which is another encryption standard that we can use, which is 168-bit encryption. We also have AES, aka Advanced Encryption Standard. AES comes in a 128-bit, 192-bit, or 256-bit encryption method, and those are standards that we use today.

3DES and AES are more appropriate these days than DES. DES is probably one that you're not going to want to use these days, simply because it's not very secure in terms of capabilities anymore. We use these encryption algorithms to hide our data, and once we do this, it involves the use of a key.

When we talk about keys, we first have to understand that there are two different types of keys. We have a symmetric key and an asymmetric key. When we talk about symmetric keys, we use the same key for encryption and decryption. For particular algorithms, a symmetric key can be used for real-time exchange, because it can happen very quickly.

For example, let's say we have a VPN tunnel, but first, we should begin with having a computer sitting behind our VPN gateway. The VPN gateway could be a Cisco firewall, and then out on the internet, we have some type of connectivity. Maybe there's another Cisco firewall on the other end, and we have a server on that side we want to talk to from our PC.

We're going to have the firewalls build encryption for us, so they'll put up a

tunnel between the two of them, and they will do real-time encryption on our data. As we send that data, it's going to be clear text or precise data, but once it's going to hit the firewall's interface, we are going to have encrypted data.

That encrypted data is then sent real-time across the network, and once we get to the firewall on the other side, the data is going to get decrypted using the symmetric key, and then we have clear data again when we talk to that server. This is what we typically use.

If we're talking about asymmetric algorithms, we would be talking about something like RSA, aka Rivest, Shamir, and Adleman algorithm, or only the RSA algorithm. We use a public and private key pair in this case. Public and private, and it's essential to understand because there is a difference.

Asymmetric algorithms are not necessarily a good algorithm to use for real-time encryption, but they are good to take some data, encrypt that data, and then later on, maybe use it to decrypt it. We can also use RSA algorithms for authentication. That's the high-level overview of confidentiality.

We can provide confidentiality, using different encryption algorithms, and we're going to go into more detail on the cryptography concepts shortly, but for now this is just a high-level overview of what confidentiality provides for us.

Chapter 2 Integrity

When we talk about is integrity, more specifically, data integrity, the idea is to make sure that data has not been modified. We have to be able to validate the integrity of our data. Usually, we use some type of hash function to verify the integrity of our data. Mostly, we have two significant protocols that we would look at. We have Message Digest 5, aka MD5, and we have SHA, aka Secure Hash Algorithm. MD5 is a 128-bit hash, and SHA is a 160-bit hash if we're using SHA-1, but there are other SHA methods that we could use.

Imagine that we have our data, and we want to validate that this data has not been modified. We want to verify the integrity of the data. So, what we can do is this. We take the data, and we make a copy of it. We take that copy of the data, and we're going to run it through a hash algorithm.

You might think of it as a kind of funnel, and that funnel could be MD5 or SHA. Whatever hash algorithm it is that we're using, next we run it through that hash algorithm and once done, the date will come out with this big mangled bunch of almost nonsense. This is because it’s a hash, something that we can't read. Next, we take that hash, and we attach it to the original data, and now we've got this hash attached to the original data, so now we can send that data and on the other side of the network. There, they can verify the integrity of the data. On the other side of the network, we get this data that comes across, and that data, whether or not it's encrypted, has a hash. The other side knows the algorithm that we're going to be using, so they already have the key. We will discuss later how the key exchange happens, but for this example, let’s imagine the other side already has those keys.

We use MD5 or SHA on the other side of the network too, and what happens, is that they take the data, make a copy of that data, run the copy through the hash algorithm, and then they take the hash that we sent and see if it equals to the hash that they've generated. If this is the case, then we can verify that the data has not been modified while it's been in transit.

Another way we can describe how this works is this. Imagine that you are shipping a package to a friend. Imagine that we have this package, and we box it up, and we're sending it to our friend, and we're going to send this through a shipping company. Regardless of the company we use, there will be transport mechanisms, and once we hand the package over to this transport

method, we don't have any control over it. We can't see the package, so we don't know what's going on with the data. Therefore, what we do is that we take the box, and we put the box on a scale. Our scale tells us that it weighs 10Kg, so we know how much it weighs when it leaves. We print out a shipping label, and the shipping label has the “TO” address on it, and the “FROM” address, and it also has a weight on it.

So now, we can imagine that this weight is the integrity hash. We have a weight, and that weight tells me that it's 10Kg. We have fixed that with a sticker to the outside of our box, and then our shipping guy comes over, and he picks it up, and it's in transit. Now it gets to the other side, and they deliver this box. I sent this box to my friend, and my friend calls me and says that he has got the box, so it made it there, but making it there is not enough. I need to know that integrity is still intact. So how do I do that? Well, my friend is going to look at this shipping label, and it's going to weigh 10Kg.

He is going to take that box that I've just shipped, and he is going to put it on his scale, and he is going to look to see if it's 10Kg. If it's not, then he is going to know that something's happened to this package in transit. Let's say that the box when he receives it, only weighs 5Kg. Well, that is a big problem. We've lost 5Kg along the way, so at this point, we would understand that we have had some kind of an issue while it was in transit, and it is no longer a valid package.

If we're talking in terms of data networking, and it’s a VPN, and that integrity hash fails, then we're going to discard the package. We won't read it; we don't want to have anything to do with it, because it is not what we expected it to be. That's integrity. Providing data integrity just means that we have a way to verify that the data has not been modified.

Chapter 3 Availability

Availability, putting it simply, if our systems are not available, then the business will not work. It's that simple. Availability has to do with making sure that the devices that we have are available in the network. That means that we have to maintain our hardware, and we also need to have a plan for failover to some degree for high availability to provide redundancy.

There's a lot that goes along with availability. It's essential to do things like upgrades and making sure that we follow a vendor's upgrade path to make sure that we are using a stable software, something that doesn't have any threats associated with it.

For example, Cisco is excellent about putting out information on current threats, letting us know that there is a problem with a particular software version. So, making sure that we follow our system upgrades as needed.

Making sure that we have the amount of bandwidth that we need in a network also lends itself to availability. Even necessary to make sure that we are preventing bottlenecks. If we get too much data and we start having traffic drop, and the network is no longer available, then that becomes a problem as well.

To put it simply, availability means that we make sure that the network is available, otherwise we are unable to conduct business. Later on when we start talking about some of the threats on a network, some of these threats will target the availability of a network.

An example of that would be a denial of service attack. A denial of service attack would be trying to prevent the network, or devices on our network from providing the necessary services. Now that we've covered a good overview of confidentiality, integrity, and availability, let's move on and look at SIEM technology.

Chapter 4 Security Incident Events and Monitoring

To discuss SIEM, aka Security Incident Events, and Monitoring technology, you first have to understand a few things about logging. Logs are a crucial part of a secured network because logging data for troubleshooting or for doing policy compliance auditing it's advantageous.

In a secure network, we're always going to have some type of policy, and we're going to have to audit that policy to ensure that we're meeting the criteria of that policy. Thus logging is what enables us to do this. We can use our logging data also to recognize an attack that's in progress, to see when an attack started, and most of our Security devices support the ability to logging. For example, let's imagine that we have a firewall, and it can send log data to a few different places. It can send it to our CLI or command-line interface, using console cable or TELNET, or SSH, aka secure shell.

We get logging messages, but in most cases, that's not that we want unless we're in the middle of troubleshooting. We can also send logs to the buffer that would be an internal storage location. If we were to send data to the internal storage location, some considerations have to do with how much storage space we have. If that buffer gets full, we're going to have to overwrite some of the old logs. We can also send it to Flash, which is another storage location, but here too, we have to deal with storage. The other two options that we have are remote options.

One option would be to using a Syslog server, and the other option would be using SNMP or Simple Network Management Protocol. These would be remote options where we would send that data off to a server, and we often call them a log collector server. That could be a log collector that is taking your data in, and displaying it in a specific Syslog server. In addition to that, we have our Firewall, and it's able to send data. Still, we also have other networking devices, such as switches, routers, or other security devices such as Cisco ACS or Cisco ISE.

Well, Cisco ACS is not supported anymore, but I know for a fact that many companies still using the ACS server. Either way, we have many devices, but most of these generally have the same type of logging capabilities. Yet, one of the things that you need to consider with these logging capabilities is that the format is similar, but the log messages are usually different. So, for

example, the Cisco ASA Firewall is going to have some logging messages that are going to be very different than your logging messages that you would get from Cisco ISE, or a Cisco ASR router or a Cisco 9300 switch. That means that you're going to have multiple files, or at least you're going to have quite a few different types of messages in this Log collector server.

Instead of organizing all those different kinds of outputs, we can use SIEM technology to collect all that data from various Syslog services or SNMP services. Then the SIEM technology will correlate that data and give us the ability to act on that information. SIEM technology would provide us with the ability to log all data into a single location and then act on it. SIEM does data aggregation, and capable of alerting on certain things. We can generally set up different types of filter rules, and so on. We can use that data to check compliance.

As discussed the how and when once we have a policy we need to look at, our logs can ensure that we are compliant with that policy. This could be internal policy, or this could also be federal regulations and so on. But SIEM devices help you do other things too, such as retention. SIEM technologies are generally designed to help you retain that data and by retaining that data, you can keep it for years. You can store it at another location, but SIEM is designed to do this for you, so you have the ability to do forensic analysis by looking at an attack after it happened for example.

When it comes to SIEM solutions, Cisco used to have a device called Cisco MARS, that used to be their SIEM device. This was a device that they end up picking up from Protego networks back in 2004 but if the project was discontinued. Many vendor-neutral SIEM technologies that are developed these days are designed by companies that want to support multiple vendors. One great solution out there currently is called “Splunk”. To find out more about Splunk, you visit https://www.splunk.com/

Chapter 5 Security Terminologies

If you look at the news, there are at least once a week we hear about a big company that has had some kind of a security breach. What we learn from that when we step back, and we look, is that there is not a single industry that is exempt from the threats that are up against. It doesn't matter if it's the entertainment industry or if it's retail. When you think about attackers, they are not just individuals anymore. Instead, there are teams of hackers today, and they act as organized crime does. They use different techniques to embezzle money, to steal money from people, to make a profit. When we are talking about the landscape of our threats today, you need to understand some terminology that's going to help you to have a better overview of what's going on. Therefore, let me share with you some known vocabulary.

Vulnerability

First, let's begin with a term called vulnerability. Vulnerability is the weakness that compromises our security. You might hear that Windows is vulnerable to certain things. So there's a weakness. There's a problem with a particular application that runs on Windows that causes Windows to be vulnerable to some kind of a threat.

Exploit

Another term that we use often is called an exploit. An exploit is what is used to leverage the vulnerability. So, if we have a weakness, the exploit can be used to capitalize on that particular vulnerability, and usually, that's going to come in the form of a tool. There are several tools out there that are used by attackers. There is a long list of security tools that are used, and attackers use many of these. Attackers will use these tools to capitalize on a vulnerability in a system. This is why it's essential to pay attention to our updates that come from our vendors, because once vulnerabilities are identified, generally they're patched, thus that's why it's important for us to keep our systems up to date. That lends itself to the term availability that we talked about earlier.

Threat

The threat is the circumstance, or the event that has the potential to cause harm. For example if there's a shark swimming in the ocean, is it a threat?

Well, not if you are standing on the beach. It's not a threat to you.

If you are swimming in the water, and there's a shark swimming near you, then that shark is a threat to you. It's the circumstance that you are in. How do we put that in terms of our networks? Well, if we have data in motion and that data in motion is traveling across a public network, then we're in a circumstance where we could have somebody capture that data, and steal that data from us. So there's the threat.

It would be vulnerable if it's not encrypted, and the exploit would be a tool that they could use such as a packet capture tool or a packet analyser. These things are all work together.

Risk

Risk is the likelihood that a threat using a particular type of attack is going to exploit a vulnerability that you have. These are how they all come together. The risk is the likelihood. Back to our example, where there is a shark swimming in the water. Is that a threat? Well, it depends on the circumstance.

You have to ask the question: What's the likelihood that it would be a threat to you? Well, if you are in the water, then there's a higher risk. If you are on the beach, then there is a lower risk.

Back to our example of data that is in motion across a public network, what is the risk to that data? Well, if the data is encrypted, the risk is much less. But, the risk is higher if the data is clear text, because the threat is that the data could be captured, and it could be modified, or it could be stolen. The vulnerability is that it might be precise text data, and the exploit would be a tool that could be used, such as a packet analyzer.

These things all come together: vulnerability, exploit, threat, and risk. These are all terms that help us to understand the landscape of network and data security. For example if we look at an attack such as a denial of service attack. Well, the denial of service attack, that's a threat to you if your servers are public servers, so then we can look at what the vulnerability is, what the exploit is that people use, the risk level to you.

But when we look at the exploit, and we look at the vulnerability that helps us to identify what type of security features we would implement to protect

against these threats. That's a look at some standard security terms that we use, specifically for terms that help us to understand the landscape of our threats that we have to deal with. Let's go ahead and see how we identify some common security zones.

Chapter 6 Security Zones

Now it’s time to look at some of the most common network security zones. When we talk about standard security zones, there are seven zones. In terms of the common areas that we have to protect against are as follows. First, let's look at a public access zone.

Public access and Common zone

Public access zone is a zone that we own, we have access to, we have control over, and it's a zone where we would probably see something like our web servers. Within that public access zone, we would have a point where it has connectivity out to the internet, where we don't have any control over. That's where all the attackers are, that's where people on the outside are, which is the internet.

Therefore, any time we have connectivity between a public zone and the zone that has public access like our server from, we need to consider that zone. That's going to be a common zone that we work with and provide connectivity between. Within the common zone, most times, we have a firewall where we'll be implementing static network address translation.

As we're providing static NAT, we would also be doing access control to make sure that only the services that we're going to be offering to the public, such as web services, are going to be visible from the outside, or the public zone.

Operations zone

Another common security zone that we're going to deal with is the operations zone and the operations zone, if we were to tear that down and look at in a little bit more detail, the operations zone would be where we have our day-to-day activity. Things that we are doing in our day-to-day work. It's where our operational virtual local area networks or VLANs would be. It's where we're going to have different subnets and other general operations. We could also look at that operation zone as the management side of the network as well.

Restricted zone

In some cases, we need to limit the impact of infected hosts, and so we're

going to put those into a restricted zone. That zone might need some protection, so we might need to break that part of the network off. The restricted zone that's generally going to be where we have our databases and our compassionate information. So, we're going to want to protect that restricted zone from denial of service attacks. This is where our mission-critical, sensitive data is going to be seen. That's a standard zone that we will protect, but when we look at this, and we just step back and think of this from a high-level perspective, we could zone off the network into several zones.

When we step back, and we look at it like this, we're going to have to identify what's called a zone interface point, and that's the point where one zone touches another zone. Generally, at the zone interface point, we're going to have a device that will have policy implemented on it.

Therefore it’s important how we implement policy controls on our network devices, because some of those network devices acting as a zone interface point. Still, some of those devices not acting as a zone interface point. Some of those devices may be inside of a zone, and there may be some threats that we have to deal with there, where the risk is high so we want to implement certain features to help mitigate some of the threat that we have there.

Regardless, just stepping back and looking at a common network security zone, we do see some zones that we're going to deal with probably a little bit more than others. For example, the zone interface point between the public and the public access zone, the highly restricted zone and the operation zone. These are zones that we'll deal with maybe a little bit more than others, but in the end, it doesn't matter what zones we have. It's important to be able to identify those zones, and ensure that we have some type of policy that controls connectivity between those zones. In summary we've started talking about that CIA or confidentiality, integrity, and availability. We talked about how confidentiality is the ability to hide our data, to hide it in transit, to maybe encrypt it on disk, whether it's data in motion or data at rest.

Confidentiality means people that aren't allowed to read it, can't read it. We also talked about integrity and how we use a hash mechanism to make sure that data is not tampered with. We also need to provide the integrity of the data that we are transmitting. We also talked about availability, and we know how important it is that our data is available to us, and our network is available to us. We also need redundancy, we need a backup plan, as they all

play a part in availability. In addition to that, we talked about SIEM technology. When we spoke about SIEM, we talked about logging. We know how vital logging to us. Logging from multiple devices can be hard to filter through. It can be a lot of work, and because of that, we want to make sure that we have some type of a device in our network, especially in more extensive Enterprise-style networks where we can aggregate all that data into one place and then correlate it.

Splunk is an excellent example of a vendor that provides a product that does that. They're not the only ones out there, but they are a well known SIEM technology. In addition to SIEM and logging, we looked at four different security terms that were going to be vital to us. We talked about the landscape of threats, and we talked about how we do have things in the network and how it all works together. There are vulnerabilities, there are threats, and there is risk. We talked about an example of a shark in the water. Is it a threat? What's the risk? We put that in terms of a data network, and hopefully, you have a pretty good understanding of what a vulnerability is.

That's the weakness that we have. The exploit is the mechanism that's used against that weakness. The threat is the circumstance that we're in or the event that takes place that has the potential of exploiting a vulnerability, and we also talked about risk. What's the risk that this could happen to us, or the likelihood that this could happen to us. Those were those four important security terms that help us understand the landscape of threats, and we wrapped it all up with a look at common security zones. When it comes to common security zones, network zoning, in particular, zone interface points, and a little bit more on a secured network architecture there, it is imperative for you to understand as you advance in your careers. Now it’s time to look at common security threats.

Chapter 7 TCP SYN Flood attack

To begin with, we should define what DoS and DDoS stands for. Starting out with that term DoS, the term stands for denial of service. A denial-of-service attack is a very nasty attack. DDoS is even more offensive; it's a distributed denial-of-service attack. Imagine that we have servers inside of our network and that server is offering an email service. We might say that is an SMTP server.

So we've got our SMTP server and on the outside of our network, out on the internet, somebody is going to be able to connect into that email server using TCP port 25. That's the SMTP port so we would establish a connection in from an external mail server to our internal mail server. That external mail server could pass us email messages that are destined for our internal users.

All of that sounds good, but in a DoS attack, we could have a TCP SYN flood attack. What we're talking about here with a TCP SYN flood attack is an attacker on the outside initializes an inbound TCP connection that would have to go through a 3-way handshake. The 3-way handshake is starting with a SYN, followed by a SYN/ACK, followed by an ACK message.

If we're on the outside, making a connection to the inside, it would be a SYN (stands for Synchronization) coming in, then a SYN/ACK (stands for Synchronization & Acknowledgement) message should be going back, and then an ACK in the opposite direction to establish the 3-way handshake. But, what happens when this is an attack or a TCP SYN flood attack, is that the inbound TCP SYN message, that first message that comes in, it would have a valid destination which is going to be our server.

Yet, the source address from the attacker will not going to be his real IP address. He is going to use a fake IP address. Our email server is going to send a TCP SYN/ACK back to the destination, back to the attacker, and when that packet comes back, it's going to an invalid host. A host that does not exist in most cases.

That means that the final part of the TCP SYN flood attack or the last part of a normal 3-way handshake can't happen. We do not get an ACK back, which means that the server is sitting there and just waiting and waiting and waiting, and nobody ever sends an ACK (stands for Acknowledgement) back, but he's

got this service that he's waiting for. He's got a connection that's opened, or at least it's halfway opened. We call this an embryonic connection or a half-open connection, and that can be used as an attack to consume the resources on the server.

Therefore, if he consumes all the resources, if this server is too busy waiting and waiting and waiting for devices to complete that 3-way handshake, then he's not able to service authentic users, which would mean that we have been denied service. The TCP SYN flood attack is very common, but it's not the only type of denial-of-service attack.

Chapter 8 Ping of death attack

There is another attack that we used to see in the past that's not very popular these days because most systems have been patched to prevent it, but it's called a ping of death. Here's how it worked. Let's imagine that we have our gateway or internet-facing router, and that is our only means out of the network, so this device has to be available to get internet access. We discussed CIA before.

This is where availability comes in. This gateway has to be available in the network. But, if on the outside world, somebody finds the IP address of our gateway and they send a ping message to it. By the way, in case you are not familiar with PING, it uses a protocol called ICMP or the internet control message protocol.

ICMP has an ICMP echo request message that's followed by an echo reply. It's not stateful, instead, it's a simple way for you to go and ask a device: “are you there?” and have a reply back to say yes, “I'm here”.

With that in mind, the ping of death was used to flood gateway devices or host devices. Attackers would take the packet and create a fragment, so a fragment by chopping up the packets. They would take that fragment and set the size of it as high as it could be, which in IP version 4, it's a 65,536 bytes.

That's a lot for a system to handle. So what would happen, is we would send this ICMP with a fragment size of 65,536 bytes and when the gateway or the host device, receives that, it tries to allocate buffers to handle that message, because it's a fragmented packet.

It's got to store these packets and put them back together. So what would happen is we would consume all of its resources. It would try to allocate memory. Eventually, the system would crash and the attacked device would reboot, causing a service to be denied.

ICMP IPv6 ping of death

Recently we've seen another type of ping of death, but instead of targeting ICMP for IP version 4 addresses, it's targeting ICMP for IPv6 addresses, and it's called the ICMP IPv6 ping of death. But the ICMP IPv6 ping of death is based upon a very similar concept as ICMP IPv4 ping of death. With IPv4, most of our systems are patched these days, but if your system were not repaired and could not handle the continuous ICPM echo requests, then that gateway would no longer be available, which means that you would be cut off from the outside world.

DoS in malformed packets

There is also another type of denial-of-service attack that comes in the form of malformed packets. When we're talking about malformed packets, we're talking about a protocol such as SNMP or simple network management protocol, or perhaps DNS that we use all the time on the network.

Attackers typically use UDP based protocols to malformed them, so when an end system such as a PC tries to interpret the packet, it can't handle it therefore, it crashes. This is just another form of a denial-of-service attack. So far we've been talking about attacks that are designed to prevent you from offering a service whether it’s an email service or a gateway service. We've also only been talking about a single machine launching these attacks against a system. These are DoS attacks.

DDoS Attacks

When we have an attack that is initiated from multiple machines at the same time, now we're talking about a distributed DoS attack. In a distributed DoS attack, the concept is simple. You have a host machine, let's say it's serving Google.com. So, it's the Google.com web server, and then we have a couple of devices out on the internet, such as Attacker1, Attacker2, and Attacker3. Each of these attackers launches a TCP SYN flood attack all at the same time to Google.com. Google.com sends a SYN/ACK back to each of these Attackers. But, the response is never going to come back to those machines because they were all spoofed addresses. Necessarily, they're not the real source IP addresses. These attackers are rewriting the packet header so that Google doesn't know who they are. They use addresses that don’t exist, and

there's a whole list of these that are not used on the internet.

To find unused IP addresses, you could Google that and look for a list of bogons. Either way, that's the difference between a DoS attack and a distributed DoS attack.

Chapter 9 Botnet

There's another attack that's very common today and that attack is called a botnet attack. Botnets can come in the form of a denial-of-service attack, but here's what a botnet is.

The Botnet name comes from the words; roBOT and NETwork, and we should categorize botnet as a type of malware. Once a botnet affected a computer, it’s called now a bot, and it is under third party supervision. You might reason that you would be aware that your machine is affected, but I am here to tell you otherwise.

The fact is that there are millions of botnet affected computers and other networking devices right at this moment, yet to detect any is very difficult. As for the end-user, all seems as it should be, no issues with connecting to the internet, neither problems on the PC, while it might be already turned into a zombie. Zombie, aka a bot.

More compromised workstations become bots, larger and more powerful it can become the actual Botnet or Robot Network. What’s happening is that each of these zombie computers is now called home, called a C&C Server, aka Command & Control.

C&C is software, yet it’s running on a Server; therefore, people refer to it as a C&C Server. So once an end device is a zombie, the attacker or whoever controls the C&C Server now able to control all the bots and use them as he or she would want to. How does it work? Well, first, let’s look at the origin of the Botnet.

When it comes to a botnet, it is so powerful that it doesn’t necessarily require to be clicked on, but you can find those forms of botnets too. Due to its malware type, Botnet can be picked up from social networking sites, e-mails, free software downloads, youtube videos, free movie downloads and so on.

Equally to Spyware, it can be attained from many sources, and once your computer is affected, it could spread around to all your devices that might be on the same network as your infected device.

For example, if you have a workstation, a laptop, an X-Box, and a mobile phone on your home network and one of them is affected by Botnet, all your devices can become part of the same Botnet. It can spread itself at any point.

For example, when you download free software from an untrusted source, it might contain a Botnet that would be hidden under a Trojan type of virus. Yet, it could be in another form such as an email link. For example, you receive a dodgy e-mail saying that you have been chosen to won x amount of money, so you must click on the link to claim your money.

Again, while you would click on that link, you wouldn't realize that the Trojan is already installing itself on your computer. Therefore it's hazardous and nearly impossible to know if your PC might be already a Zombie.

Another infected media could be a USB Stick, or nowadays, even cheap smartphones bought from China can contain Trojans that would spread around to other devices to expand the robot network.

Imagine that a torrent movie is effected with a Trojan that would contain a botnet, and 3000 people are downloading it every day for the next three months. Eventually, those 300K+ computers would become a bot for a particular robot network.

Yet you might think, how on Earth would all those bots connect to a C&C Server? First of all, 300K+ computers to be on the same botnet is an average number.

Experts have compromised Botnets previously that were large as 30 million zombie computers called BREDOLAB that were also running on under an alias of OFICLA. This was a Russian botnet, but it has been compromised. The fact is that we don't know, at least not sure how many Botnets are out there.

So back to the victim's computer, once the botnet installs itself, called a BOT Binary, it would still have to look for a way to connect itself to the C&C server to communicate to each other and exchange messages. BOT Binary can contain a hardcoded IP Address that would advertise it's details out to the internet, so the C&C Server would find it's bots. Still, there are other methods too. Another common way would be that a particular Domain name is written into the BOT Binary that would be advertised out to the internet to find it's master C&C Server. Either way, once the Zombie computer registers itself to the C&C server, it will become a BOT officially, and the Robot Network Army begins to grow.

There are also good intentions for someone creating and using such Botnets.

Yet, there are very few as we know of. In certain countries, some websites are blocked; therefore, a few communities are using Botnets to access the information that their government wouldn't allow them to view according to their law.

Botnets are used mainly by the bad guys, but to be more specific, large Underworld Cyber Criminal Organizations. Similarly to Spyware, once your computer becomes a bot, it could forward all sensitive information to its master – C&C Server that might be usernames, passwords, or bank account information, but the primary purpose of the Botnets is deeper than that.

Some people would only build Botnets so that they could sell it to Cyber Criminals. More massive the botnet, the more value it has. Of course, certain botnets would contain only bots from the US or Europe, so those would be a little cheaper.

Large Botnets that have bots all over the world in different continents are more expensive. A botnet that would contain a C&C Server and 50-100 bots would be sold between $200 - 800, yet it all depends on the locations of the bots.

Taking this further, Cyber Criminals have multiple botnets, each would contain 10K + zombie computers, and they would let them out for an hourly fee, or daily fee. It would depend on the requirements, as well the quantity of the bots, and their location, but an average price for 5000 bots with C&C Server for 1 hour is around $100 to $1000/Day.

When it comes to a botnet of 5000 bots, not all 5000 zombie computers can be used at the same time, because some of them might be turned off. Still, I wanted you to understand the pricing when it comes to a marketplace.

Back to the purpose of the botnets, some organizations would use it to create a DDoS (Distributed Denial of Service) attack against a particular company, perhaps against their competition, but also seen botnets used for revenge of an ex-employee. Either way, botnets can be used for attacks, but mainly used for financial gain such as Bitcoin mining.

Bitcoin mining is prevalent, but to mine Bitcoin, you must have a tremendous amount of CPU power combined; therefore, large botnets can be perfect for this exercise. This process is also known as Silent Bitcoin Mining. This must be controlled accurately because, for Bitcoin mining, all bots would use

pretty much 100% CPU. So, whoever would sit behind the C&C Server would manage the usage of bots, so the victims wouldn't realize that silently, their computers are mining bitcoins.

But hold on a second! Who is sitting behind the C&C Server?

Well, as I mentioned, all bots are centrally controlled by the C&C Server. Due to the centralized coordination to compromise such a robot network, the source could be identified. Yet the fact is that the Bot-master would be very careful and probably only log into the C&C Server once it's fully secured.

When it comes to the C&C Server location and a skilled Bot-master, it is guaranteed that it would be hidden behind a multi-layered network called TOR. TOR network would allow the BOT-master to be anonymous. Therefore it would remove all traces of his or her identity, resulting in the BOT-master to be untraceable.

How to avoid your computer to become a Zombie?

The answer is simple – back to basics! Do not download software from untrusted sources; even if the software is free, you must make sure that you are getting it from a trusted source. Downloading torrents like movies, music, or video games, I recommend you do not do it, due to the potentials for those items might be affected is indeed very high. E-mails that advertising things that are too good to be true, DO NOT OPEN them, period! Straight away report them as spam. Do not even open emails like that because the attacker might be monitoring who is opening those e-mails. Once the attacker sees that specific email opening rates are high, he would make sure to send a similar email even more frequently.

Your computer must not remember your username and password/s. Similarly, in case you buy a new laptop or desktop computer, you must change the passwords. Furthermore, be careful and be reasonable with the information presented to you.

For example, on social media, you may see things like you can win $1Million, and all you have to do is click on the link to register, please use common sense and do not click on untrusted links, and certainly do not provide your personal information.

Once again, do not click on anything that you are unfamiliar with, especially

weird programs that would supposedly help you achieving thinks like hack into someone’s Facebook Account and thinks like that.

Instead, you should purchase an Antivirus and update it regularly. Moreover, you should install a Firewall, even if it’s a virtual Firewall, because it would still help you identify if you are affected.

Next, you must always run the latest operating system, especially if you have Windows. They usually do upgrades within their software each time they find a vulnerability within the previous version; therefore, you must upgrade to patch those vulnerabilities.

Botnets very frequently feature in distributed denial of service attacks, and it makes sense because it gives the attacker a way of leveraging their attack by using infected machines to multiply the effectiveness of their denial of service efforts.

In summary, here's the way a typical botnet is put together. We have an attacker, and often, when we talk about botnets, the attacker may also be referred to as a bot-master. This is the person who is responsible for corralling all these infected machines that form part of the botnet. The attacker communicates with a control server, and a control server is also referred to as a command in control or C&C server.

This is the point that is going to do all the planning of the individual infected machines that run in the botnet. The C&C server is a critical part of the infrastructure.

In large botnets, there may well be multiple C&C servers. One of the reasons for this is that it gives the attacker some redundancy. If one C&C server gets taken down, say by law enforcement, others may still run and issue commands to the machines in the botnet.

But for the sake of simplicity, let's represent it in a single environment. The key to the botnet is that the control server is going to coordinate the zombies. The zombies are the individual infected machines in the botnet.

You may also see these called slaves at times, and what we're talking about here is otherwise legitimate machines. For example, PCs that have been infected with Malware are now taking instructions from the control server.

It's not always just PCs, and we see servers become infected and form part of

a botnet as well. Servers can be enormously useful to the bot-master because they tend to have very high levels of uptime and excellent connectivity as well.

This is what the attacker wants; they want to have big pipes from these zombies in their botnet so that each one can then launch attacks. Keep in mind that these are otherwise legitimate PCs, servers, used by everyday people.

They've just been infected with Malware, through one of the usual channels. Perhaps it was a popup on a website that lured them to run an executable, or it might have been delivered by email or spread on a USB stick.

There are many different ways that machines get infected with Malware. Large botnets can have hundreds of thousands, even millions of infected computers. Think about that for scale, millions of computers all under the control of one attacker.

These machines are typically used to do things such as sending Spams. We've seen individual botnets send tens of billions of spam emails every single day. To bring this back around to distributed denial of service, imagine you have millions of machines in your botnet.

If you could coordinate those millions of bots to make requests to a target website simultaneously, that's a tremendous amount of volume coming from all corners of the world.

A large scale botnet, even a botnet numbering in the tens of thousands, can be enormously operational at taking down target websites. Many of the examples we're going to look at in this book are executed via botnets.

We're going to look at the individual attack styles that are used to maximize the damage to target environments. But very commonly, those patterns then get distributed out to these zombie machines.

Chapter 10 IP & MAC Address Spoofing

Spoofing comes in a couple different forms, but spoofing is the injection of traffic that appears to be from a different source. There are IP address spoofing; MAC address spoofing; and lastly application and service spoofing. Let's begin with the IP address spoofing.

IP address spoofing

In an IP address spoofing attempt, let's assume that we have a host with the address of 10.10.10.10 and it is talking to a host with an address of 10.10.10.20. That's all good. They will communicate through IPv4 with those two addresses, not a problem, but let's add another machine in there and let's have another machine say that he is 10.10.10.30.

Well, now what could happen is the two address could start talking and the first one with the .10 address could send a packet to the second device to the .30 address, spoofing the 10.10.10.20 address, and when the .10 address replies back, he'll be replying to the original .20 address who could not even understand why he's receiving the packet. So 10.10.10.20 is sourced or spoofed. A packet is sent to 10.10.10.10, 10.10.10.10 replies to the real host at 10.10.10.20 and the real host 10.10.10.20 discards the packet because he doesn't know what's going on.

You could use to initiate a denial-of-service attack. You could spoof an address, and have all messages or responses sent to the real host, and the actual host can sit here, trying to figure out what to do with them, causing a denial-of-service attack.

MAC address spoofing

The other method is called MAC address spoofing. MAC or media access control is the hardware address of devices that are not changeable, yet they can be spoofed. Imagine that we a device called Host A. Host A sends traffic with the MAC address being spoofed of let's say, aaaa.aaaa.aaaa. He sends a message over a network switch to Host B. Host A sends a message, Host B receives it. Host B replies, but it's not really Host A’s address, so Host B, he's talking to nobody.

Chapter 11 DHCP Server & Client Spoofing

We do have some layer 2 controls to help us with these types of attacks and they are application and service spoofing. The first one is called a DHCP server attack, where I'm spoofing a DHCP server. Let's imagine that on a network, we have a switch, and we have a host that we'll call it Host A, connected to port number 1 on the switch.

On port number 2, we have an attacker who is running a DHCP server and he's not a DHCP server. He's just spoofing the DHCP server. So, when Host A comes online, it will send a DHCP request, and he's expecting to get a DHCP response. He sends a DHCP request to the real DHCP server, but frequently DHCP requests are broadcast message. It would be flooded out of every port on the switch, except the one it came in.

Eventually, the real DHCP server would get the request, and he would reply back, but what if the fake DHCP server on port 2 was to hear that request first and reply with a fake IP address. Well, in this case, this host on port 2 is assigning an IP address on the wrong subnet. He sends a fake IP address on the wrong subnet, and Host A cannot communicate with the actual network because he's on the wrong subnet. Subnet stands for subdivide networks. That could be bad, but that is an example of spoofing the DHCP server.

DHCP client spoofing

Now let's take a look at an example of spoofing a DHCP client. Let's imagine that we as switch and we have our DHCP server, which is our real-live DHCP server. We also have a client who is n attacker, so he's bad and he sends a DHCP request. Only the DHCP request is fake. How do you know it's fake? Well, remember when discussed that DHCP messages were broadcasted messages? Well, broadcast messages have source addresses, so while the broadcast address is all F's commonly written as ffff.ffff.ffff.

In this case, it would be a layer 2 all F MAC address, which would then be flooded by the switches to get the attacker to the DHCP server. The source MAC address usually would be the real machine that wants an address, but what if that source MAC address were spoofed and the DHCP server then allocates an address to that spoofed MAC address? He assigns an IP address to the spoofed MAC address. Let's say that it keeps happening, so the attacker

does another fake request again, and another address allocated, again and again.

If this happens enough times, then you've got a DHCP pool on your DHCP server that is exhausted. It's an exhausted DHCP pool. The DHCP server can no longer hand out addresses to real machines and now we're in trouble. That's an example of the DHCP server, the DHCP client spoofing. This attack is also known as DHCP starvation attack. This would be services or applications where we would be spoofing the functionality of them and causing havoc on the network. Next, we're going to talk about social engineering attacks.

Chapter 12 Social Engineering & Phishing

Social engineering is something that has been going on for a long time now, and there are a number of methods of Social engineering. For example phishing is a kind of social engineering, and if you look at the history of phishing, there are many unique phishing reports have been received as time has gone by.

What's interesting is that you're going to find so many different methods of phishing. We're just going to discuss a little bit about these and give you a bit of an overview of different types of phishing. One example would be a phone call from someone claiming to be the IT department. That's also called vishing. If someone would try to phish out some information over the phone, it’s called phishing.

Some other examples of that might be a credit card company that calls you and says that your bill is late, and then they need to verify your information. Another example would be someone calling and telling you that they are Microsoft support and your machine is infected, and they need to access to your device.

As a result, you give them access and then they either damage your machine, they install malware, or they tell you that you're infected and try to get you to pay them for a product to remove the infection.

Another example would be to take a USB drive and leave it laying around somewhere. Attackers can put malware on that USB drive and have it install silently and then from there, an attacker can gain access to your network. They then can damage or alter or do anything to your files.

Another method of social engineering would be to create a fake account on a social network and then friend someone. Then based on being friends with that person, you can learn about them and gather information. This could befriending an executive within a large company that you're trying to gain some information on, or do some intel on.

It could be friending someone in the IT department so that perhaps they give you some information. Whatever the case may be, it's using our external resources against people to gain information from them.

Another example of phishing could be that you receive a fake email from your bank that says that you inherited some money, and they want you to confirm your bank details. Attackers were trying to use people and the things that people do against them.

Initially, phishing was sent via email. When you think about that, an email message is something that we all are used to receiving, but that email could contain links, and that link could land you on bad website. That website could contain malware. That would be bad, and that's initially how it’s going.

Chapter 13 Spear phishing, Whaling & Pharming

Another example of phishing that is specifically targeted an individual is called spear phishing. The reason it’s called spear phishing is because when you spear fish, you get a single fish at a time. Now as it's evolved, there's another type of phishing that's called whaling.

The way that whaling works is that it's designed to target, for example, large companies and their executives. If you imagine, you send an email to the group of executives at Microsoft, and you tell them that they are all being summoned for a court interaction. Then from there, they have to open an attachment and agree to whatever documentation is in there, and then that spreads some type of malware within their company. So that's whaling.

Another method of Phishing is called Pharming. Pharming is similar to the two types of phishing, of whaling and the spear phishing. Only, what happens in a pharming attack is that your DNS entries are corrupted to some degree.

This could be as simple as modifying the /etsy/host file on your machine and pointing you to the wrong website, when you go to a website such as your bank, while it could point you to the wrong place. It could have a dummy website set up, and then once you enter your username and password, it tells you that the site is temporarily down.

But instead, really the attacker has just captured your password information. So that's one way of performing pharming. It could also be that a rogue DHCP server has been implemented, and that is issuing DNS. Then the DNS in the rogue DNS or DHCP configuration is pointing you to places that you really don't want to go to. So those are some examples there.

Chapter 14 Watering hole attack & Smishing

A watering hole attack is where a website is compromised, and that website has a certain group of people who visiting that website regularly, and they are the target. Thus, when they access that website, those group of people could be infected, and malware could be delivered to them, and now the attack is successful.

Then perhaps others who are not the target, and they were not part of that group, they access that same website, and things appear to be reasonable to them. It’s all depends on what the goal of the attacker is. One other method of phishing that is not as common anymore is called smishing.

Smishing is an exciting attack because it uses your SMS, your text messages to phish information from you. This could be as simple as getting you to sign up for a text messaging list to get free coupons and delivering links to you that could damage your phone or install some kind of malware on your phone or your devices.

It could be where somebody gets your number, and they send you messages and try to communicate with you to learn information about you, maybe befriend you, maybe pretend to be somebody you know and have a new phone number.

In all these cases, it is social engineering, but at the same time as being social engineering, it's also phishing. It's important to understand that not only does social engineering take the approach of using people against people, but a lot of social engineering involves this concept of phishing.

Phishing is where we're trying to gather information, by phishing for information. We're hoping to get information, but that's social engineering, and that's also phishing. I've given you a couple of different looks at how these attacks might take place, so let’s move on and look at attacks on passwords.

Chapter 15 Brute Force & Dictionary Attacks

Another type of attack that we deal with is the attack on our passwords. This is something that we all see from time to time. We know there are a number of password managers out there, and the reason for that is because we have so many passwords that we're dealing with. Attackers have many methods to get our passwords.

The first thing could be just guessing. If they're just guessing what our password is, and they can figure it out, then that's a bad thing. That means that the password that we're using is not very strong. So if they're able to guess it and we use that same password over and over, then we're in a lot of trouble. That's the first method that attackers use, guessing our passwords.

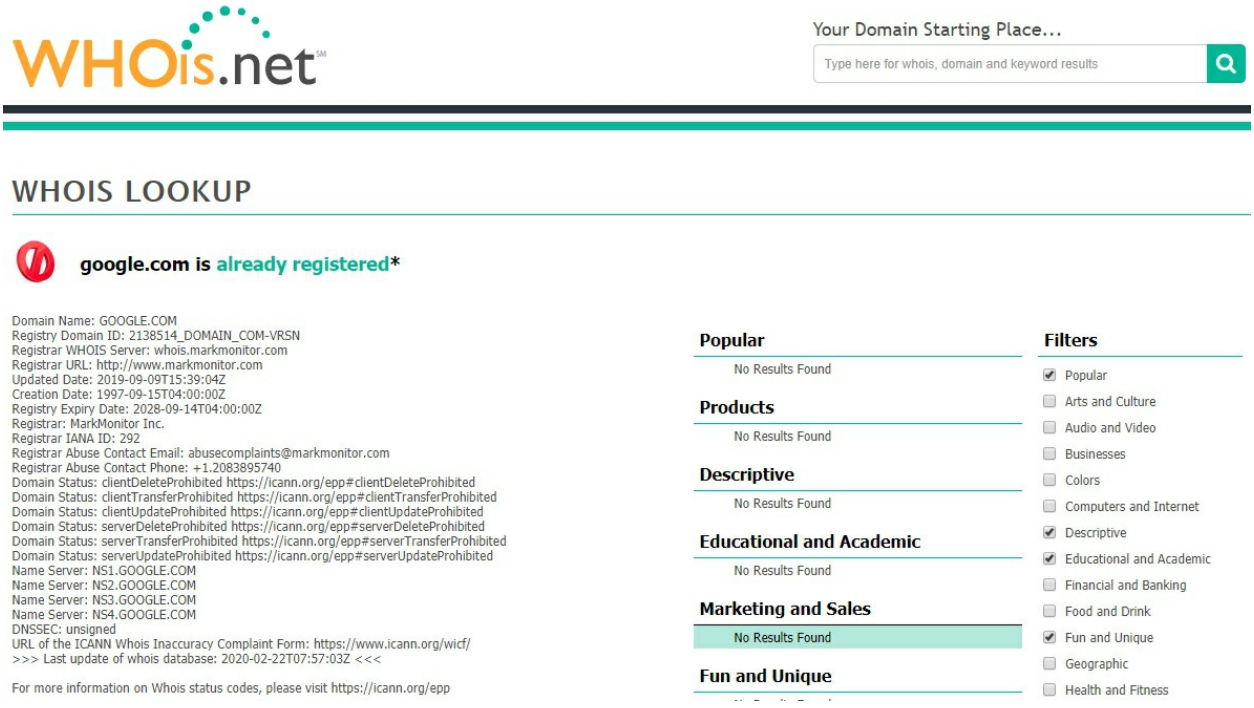
The second method that attackers use is brute force. Brute force is usually going to require some type of software, but if they have that software, they can run a brute force attack against an Active Directory domain or a web server, and eventually hope to crack your password. Sometimes they'll even do this on PDF files that are secured.

Lastly, there's another method of cracking passwords, which is to use dictionary attacks. Dictionary attacks simply use a list of words, and they try to roll through those words to figure out what your password is. They're similar to a brute force attack, but in most cases, they'll do a dictionary attack before a brute force attack. Regardless of what method attackers use to try to crack our passwords, these types of attacks are real and something we want to be aware of and protect ourselves against.

Chapter 16 Recon Attacks

Our next type of attack is a reconnaissance attack or also known as recon attack. The recon attack is all about gathering information. So if that's the case, if we're trying to gather information, there are a couple different tools that we could use to do that. Some are very simple. One of them is just doing a “whois” out there on the internet. You can visit https://www.whois.net/ and type in any website you want to have information about.

We can do a “whois” and that would go ahead and give us information about who owns a domain. For example, if you try out google.com, it will provide some information.



As you can see here, it gives me quite a bit of details about who owns the domain, who it's registered with, and things like that. What's the benefit to this? Well, the benefit to this is if you want to attack someone, you have to start somewhere. If you're going to attack an enemy, you have to do a recon mission where you would gather the intel that you need, so you know the best angle of attack. That's precisely what a recon attack is in terms of data networking.

Chapter 17 Buffer Overflow Attack

Buffer overflow attack is actually a very common attack. We've seen it happen with the SQL Server and it's used to inject malicious code in some cases, but let's take a look at a scenario how this happens.

Fundamentally we have a server, and the server is receiving input data. As the server gets that input data, that data has an expected size, so the server allocates memory to receive that data.

Now if that server does not verify the size of the data that's being received, if it doesn't do any validation on the input data, then potentially, it could take that input data and put it into the allocated memory, where there is not enough space, so it overflows the buffer and writes it into the adjacent memory. This could be bad.

If the machine accepts it, and it writes it into memory and it fills up those buffers, that could potentially corrupt the system. It could even cause that server to crash. What's the worst case that could happen? Well, the server could accept all that data, write it to memory, and that could have malicious code injected in it.

Then once that file is executed, at whatever point in time that data file is opened, well now you've infected this machine, or you've spread some type of malware, or some type of malicious activity then can takes place. So buffer overflows are horrible, but they're still very relevant.

A good idea is to keep your devices patched with vendor patches, especially things like SQL servers, but definitely something that we want to be able to prevent.

Chapter 18 Man-in-the-middle attacks

Man-in-the-middle attacks or MITM attacks, yet another horrid attacks that we've got to deal with. How do they work? Well, it's actually pretty simple. You have a couple of devices, let's say you have host A and host B and those devices are talking to each other.

Somewhere in the middle of that communication, host C appears and injects himself into the conversation. How does he do that? Well, that could come in several forms. It could be merely using an attack against a network switch where we flood the MAC address table.

Because the MAC address table is flooded, the traffic then from host A and host B and anyone else for that matter has to be forwarded out every port except for the one it came in on, which means host C is now able to hear those traffic. That's just one example of an attack that could cause a man-in-the-middle attack, but man-in-the-middle attacks are just an attack where somebody puts himself in the middle of the conversation.

Here's another example of what could come from a man-in-the-middlee attack. In this example, what we have is a couple of host machines, host A and host B, and they're talking to their default gateway or router, and their default gateway gets them out to the internet.

If you think about the process of how host A would talk to a gateway, host A is going to use DHCP. He's going to get an IP address from his gateway, so let's say that that gateway's address is 10.10.10.1. He receives the IP address 10.10.10. 10, and his gateway address is 10.10.10.1.

Any time he's going to send traffic to another network, he's going to send it to the 10.10.10.1 default gateway. But the catch here is that he can't modify the IP header and change it to 10.10.10.1 because if he did that, then the packet would not go to whoever it's destined for on the internet.

It would simply go to the gateway. The gateway would reply and we'd be done with the conversation. So what does he have to do? Well, he's got to ARP. ARP stands for Address Resolution Protocol, which is going to translate IP addresses to their relevant MAC addresses. So, he's going to send out an ARP request. That ARP is going to say, I need to know who is 10.10.10.1. The gateway is then going to reply and say, here is the MAC address of the gateway.

In the layer two header for the packet that is destined for someone out on the internet, we're going to put the MAC address of the gateway. This way, the user can then send that traffic to the MAC of the gateway. The gateway then strips off the layer two header, looks at the layer three header, and see that this is for someone that's out on the internet, so it will forward it. That's how it works.

But, what can happen is that a bad guy could send out an ARP reply and say that he is 10.10.10.1. But, is he really 10.10.10.1? No, not really, but he could change the MAC resolution, and the MAC is the bad guy. Now what's going to happen is that host A is going to try to send traffic out to the internet.

But, when host A tries to send traffic out to the internet, he's going to put a layer two header with the MAC address of what he believes to be the gateway. The only problem is that it's not the gateway; it's the MAC of the bad guy. He forwards that packet to the network switch, the network switch looks at his MAC address table, and he goes, “I know where that goes.” And he sends it to the bad guy.

The bad guy now takes that traffic, captures it, does whatever he wants to it. He can store it for later if he wants to. Then the bad guy has to forward this packet over to the real gateway so that the real gateway can send that traffic, and the actual user can get a reply back, because if the bad guy doesn’t do that, then they're going to know that he is indeed the bad guy seating right in the middle.

The point is that these attacks are bad. There are plenty of mitigation techniques that we can use, and we're going to talk about these later on as well, but that is a classic example of a man-in- the-middle attack.

Chapter 19 Malware & Trojans

Malware is something that is quite a discussed topic these days and it comes in several forms. Let's first talk about viruses. Most people probably knows what a virus is. We've got software that's supposed to protect against it, although it never seems to really be able to keep up with the viruses, but malware can be propagated as a virus by copying itself and becoming part of another program. We often see that and they're pretty offensive. Another method of delivering a malware is worms. Worms are similar to viruses. They have the ability to replicate themselves and to propagate themselves and infect the system. Worms could be used to damage the system or to provide someone access to the system.

Then we have a Trojan horse. That's, if you think back to the Greeks, it's named after that wooden horse that was used to get into the city of Troy. But basically what's happening when Trojan used is that we give you something that looks nice, but inside of it, it's infected with something that’s not so nice.

Many times these will be used to create a backdoor attack and give someone access to a system. Trojans don't reproduce themselves. They don't infect other files, and they don't replicate themselves. Usually, they're going to be spread by someone opening an email, downloading it, running the file, and then going from there. Those are the three types of malware that we see these days. Malware is generally considered or used as an advanced persistent threat or APT. An APT is a threat that is persistent, it's targeted, and usually it's targeting a specific organization or a specific person.

This could be something that takes quite a long time, but the goal of an APT is to initially compromise the host, perhaps provide escalated privileges, do some recon, and go from there. Depending on what is the goal is of the attack. But they're all done in stealth and all delivered through that Trojan horse. That's just a high level overview of a malware. What you're going to find is that there are many services that are targeting malware and advanced malware prevention but those will not be covered in this book. For now, just understand what malware is. It's software that is malicious, that's can be delivered to our machines in a few different forms.

Chapter 20 Data Loss Vectors

Data Loss is when data gets out of an organization when it's not intended to get out of the organization. This can come in many forms. A straightforward way that's been done for a long time is as an email attachment. You want to send yourself some files that you can work on at home or read while you are going home or coming to work, so you just email them to your account. Right there, we've got data that's leaving the organization even it was not intended to.

What about an unencrypted personal device? You put your email on your mobile device, and then you lose that device, and now somebody's got access to your email and those files that are located in there. Yet another way that data gets out of our organization. Another standard method is a Cloud storage service that we all use, such as Dropbox or OneDrive. They provide free accounts and gigs of storage.

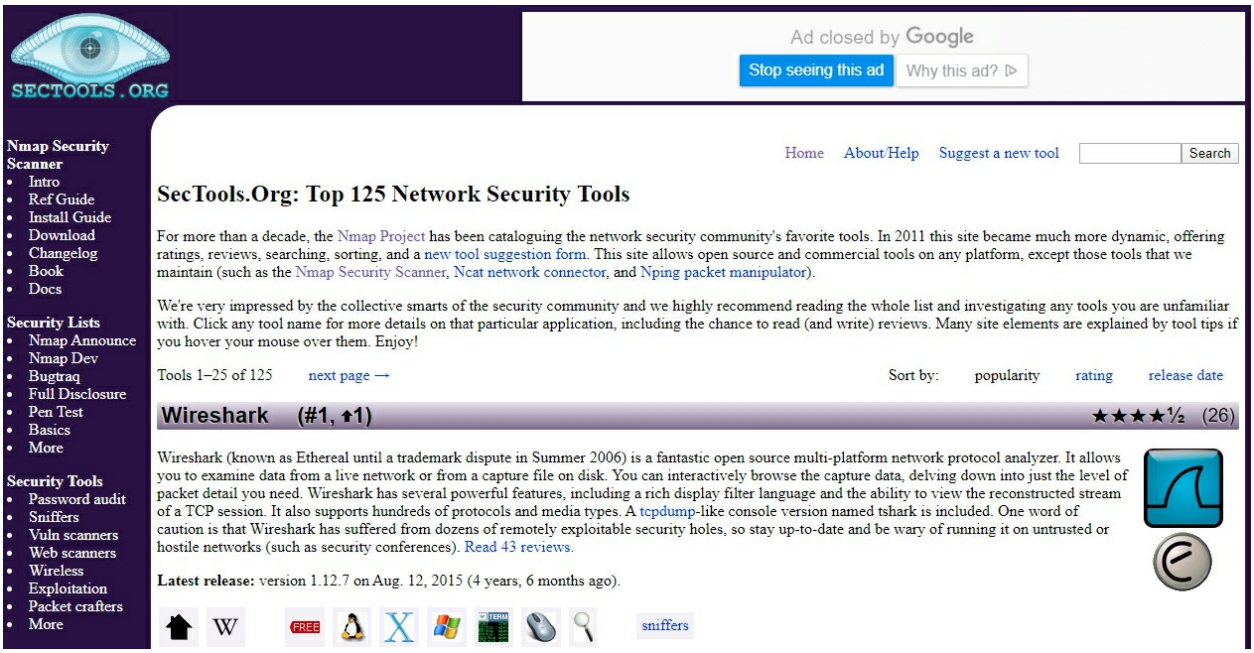
Maybe want to save something up there, so we go to the web page, and we upload it right through the web, drop it into our Cloud storage, and now if that Cloud storage provider is compromised, then we're in trouble. Then you also have USB and other removable devices. If you have a couple of USB drives, it's easy to just plug one of those in.

USB ports are located right on the front of our computers, making data loss very easy. Therefore, there should be data loss prevention mechanisms in place. There are several software that can prevent us from copying things to removable media, but this is still a real possibility and yet another way that data can leave an organization when we don't want it to.

Those are all the different attack vectors in terms of data loss. Let’s move on and discuss the tools that attackers use to compromise networks.

Chapter 21 Well Known Hacking Tools

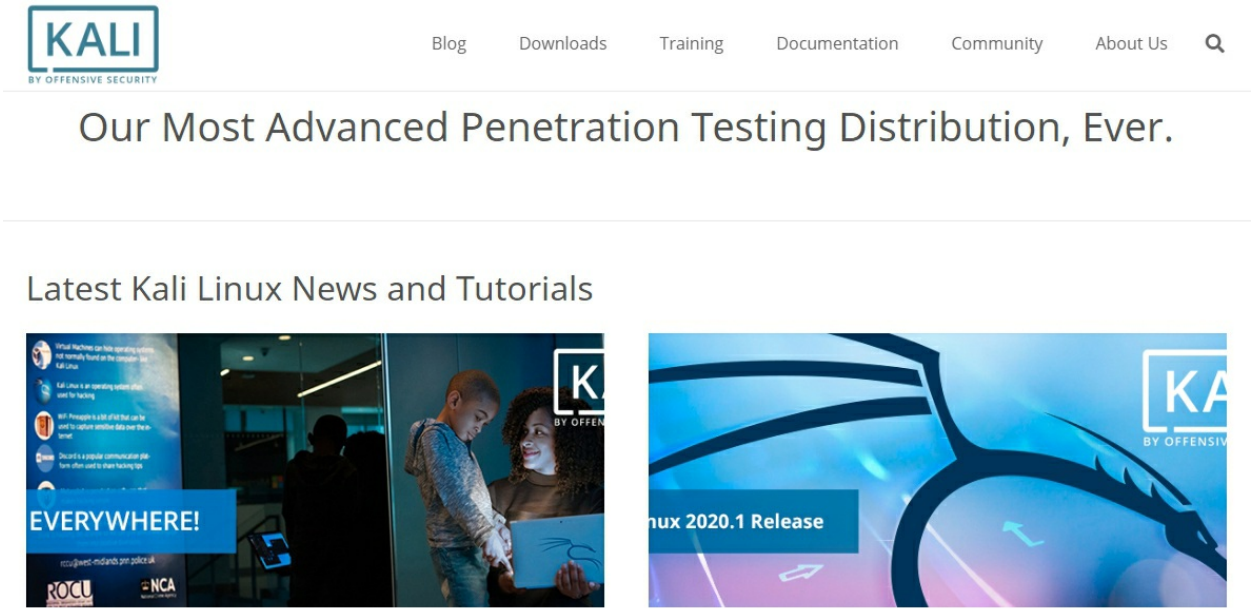
There are quite a few hacking tools out there, but if you jump on the web browser, you can visit some of the websites that I recommend you take a look at. The first one is called sectools.org.


SECTOOLS.ORG
Ad closed by Google
Stop seeing this ad
Why this ad?
Nmap Security Scanner
Intro
Ref Guide
Install Guide
Download
Changelog
Book
Docs
Security Lists
Nmap Announce
Nmap Dev
Bugtraq
Full Disclosure
Pen Test
Basics
More
Security Tools
Password audit
Sniffers
Vuln scanners
Web scanners
Wireless
Exploitation
Packet crafters
More
Home
About/Help
Suggest a new tool
Search
SecTools.Org: Top 125 Network Security Tools
For more than a decade, the Nmap Project has been cataloguing the network security community's favorite tools. In 2011 this site became much more dynamic, offering ratings, reviews, searching, sorting, and a new tool suggestion form. This site allows open source and commercial tools on any platform, except those tools that we maintain (such as the Nmap Security Scanner, Ncat network connector, and Nping packet manipulator).
We're very impressed by the collective smarts of the security community and we highly recommend reading the whole list and investigating any tools you are unfamiliar with. Click any tool name for more details on that particular application, including the chance to read (and write) reviews. Many site elements are explained by tool tips if you hover your mouse over them. Enjoy!
Tools 1–25 of 125
next page →
Sort by:
popularity
rating
release date
Wireshark (#1, ↑1)
★★★★½ (26)
Wireshark (known as Ethereal until a trademark dispute in Summer 2006) is a fantastic open source multi-platform network protocol analyzer. It allows you to examine data from a live network or from a capture file on disk. You can interactively browse the capture data, delving down into just the level of packet detail you need. Wireshark has several powerful features, including a rich display filter language and the ability to view the reconstructed stream of a TCP session. It also supports hundreds of protocols and media types. A tcpdump-like console version named tshark is included. One word of caution is that Wireshark has suffered from dozens of remotely exploitable security holes, so stay up-to-date and be wary of running it on untrusted or hostile networks (such as security conferences). Read 43 reviews.
Latest release: version 1.12.7 on Aug. 12, 2015 (4 years, 6 months ago).
sniffers

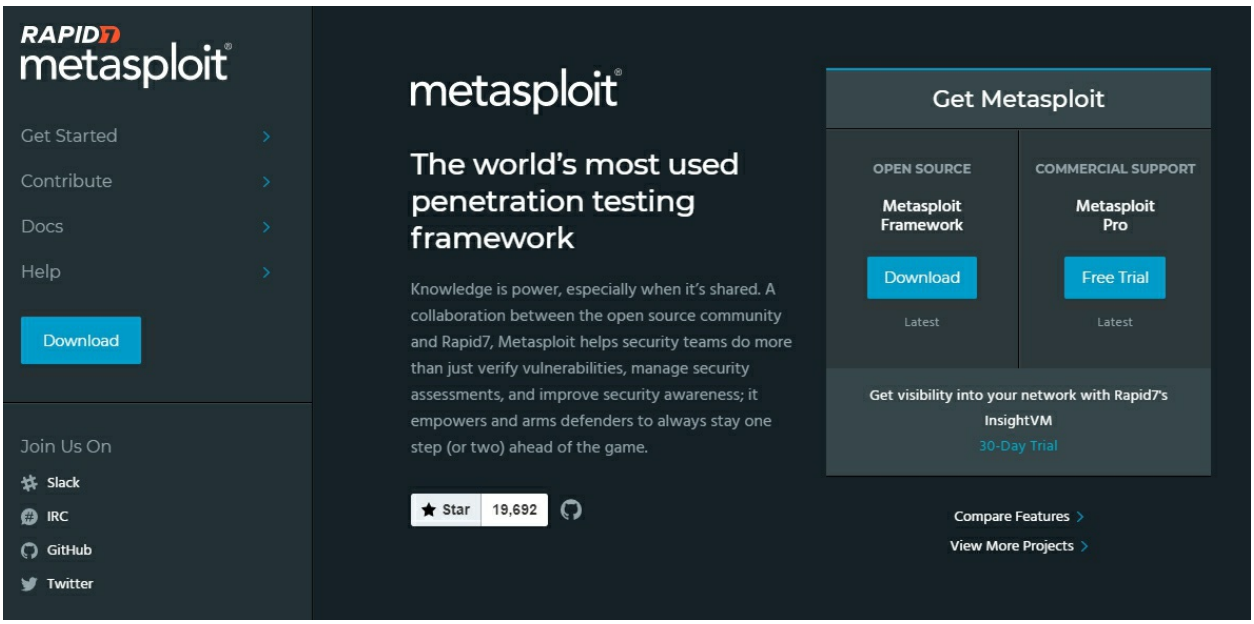
SecTools publishes the top 125 network security tools, so as you start looking through this list, you'll see things like Wireshark and Metasploit, Nessus, Air Crack, and the list goes on. Some are free, some are paid, but this is a good place to start as you look for tools that are commonly used to provide some ethical hacking capabilities. I say ethical hacking, but not everybody does ethical hacking. Yet, these are tools that you can use.

Another one is called Kali Linux. Kali Linux is a pen testing distribution of Linux. You go ahead and download the software at https://www.kali.org/. Many people like to put this in a virtual machine and that way you can just boot up that virtual machine whenever you want.

You can have Kali Linux run on a Raspberry Pi that's also a common method as well, but there are a number of tools built into Kali Linux that will help you to do pen testing and ethical hacking as well. Kali Linux has lots of tools that you'll find and can be used for free.



Lastly, there is Metasploit. Metasploit has been on the scene for a while now. It had a huge impact when it was initially introduced. It had quite a few tools in its toolbox, so you could launch an attack from Metasploit. You just select and configure an exploit in there and then each of those exploits are going to target a vulnerability of some kind of unpatched service or operating system.



You can download a free version of Metasploit at https://www.metasploit.com/. It also has a vulnerability scanner that will help you figure out which exploits you should try. That's just an example of another tool that you can use. There are three different places that you could spend a little bit of time looking at

the tools that are commonly used for hacking.

In review, we have discussed various threats to data networks. There were many different threats, but at this point, I hope that you have an understanding of these threats, at least a high-level understanding of what the threats are and that they are out there.

We also discussed different data loss vectors, and how data gets out of the organization when we don't want it to. This is something significant to us in a security world. Then finally, we referenced a couple of tools that are commonly used in attacking, so now you should have a good baseline of network security. Therefore, it's time for us to start getting into protecting against these attacks. We're going into cryptography concepts and talk about the importance of encrypting our data and how cryptography works at a very high level.

Chapter 22 Cryptography Basics

Cryptography has been around for a very long time. It's got a couple of specific responsibilities. First of all, we know that cryptography is designed to secure communications. Secure the data in transit or data that we're sending, as well as at times when data is at rest. We may have some data that's stored on a server, and we want to have that data encrypted, so if somebody were to get their hands on it, they can't see it.

We have two ends to this cryptography puzzle. We have the data in transit portion, and we have the data at rest portion, but either way, we have very similar methods of encrypting this data. Cryptography is responsible for providing the confidentiality of the data that we have, making sure that nobody sees it, and nobody can decipher what the real text is.

We also use cryptography to provide data integrity, making sure that the data has not been modified in any way or shape or form. We also use cryptography to provide the authentication of where the data is coming from, so we call that origin authentication and even for non-repudiation, making sure that the data has not been repeated and it hasn't been resent. We're not in the middle of a replay attack. So that's what cryptography is responsible for providing for.

Another element that we have to discuss in terms of what cryptography is how cryptography works, is that there is a field called cryptanalysis. Cryptanalysis is all about finding and exploiting weaknesses in the crypto algorithms that we are using. This could come in the form of a brute force attack or some other attacks that we're going to talk about later, but it's important to understand that while we do have folks that are trying to figure out how to hide the data, we also have folks who are trying to figure out how to exploit weaknesses in those algorithms. They can be used for good, but they can also be used for bad, and we see that from time to time as well. Let's move on, and look at a history of cryptography.

Chapter 23 Substitution Ciphers

The history of cryptography dates way back. There was an inscription that was found on a tomb. It was carved in a tomb around 1900 B.C., and what they found in this tomb is that there were unusual hieroglyphic symbols. It was later determined that this wasn't a form of secret writing; they weren't trying to encrypt anything, but they did incorporate some transformation of the original text was.

Later on, around 100 B.C., Julius Caesar used the Caesar Cipher as it's called, and that's what's called a substitution cipher. In a substitution cipher, each character was shifted by three places in the cipher that Caesar used. There are other substitution ciphers, in fact, and you can look at an example below. We have an example of a substitution cipher here.

What you'll notice is that along the row, we have the alphabet from A to Z, and then in the row right below that, we have the very first couple of letters there. We use the word cipher, and that shifts the letters for us.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
C	I	P	H	E	R	A	B	D	F	G	J	K	L	M	N	O	Q	S	T	U	V	W	X	Y	Z

So we can see the cipher, the A would be the equivalent to the C, the B would be the equivalent to the I, the C would be the equivalent to P, D would be equivalent to H, E would be the equivalent to an E, and F would be the equivalent to R and that would spell the word cipher.

From this point on, with the remainder of the alphabet, we could just continue on: A, B; we already have C in our cipher so we're going to go D and so on. You can see how this works with the rest of the letters. There's 26 letters and we've got some of them adjusted and it's shifted to a degree. Then we've taken some text and we've encrypted it. So, as an example, let’s see I want to encrypt the word “CRYPTO” The first letter of C would be a letter of P. The second letter of R would be an Q. The third letter of Y, would be also a letter of Y. The forth letter of P, would be a letter of N. The fifth letter of T would be also a letter of T. And the last letter of O would be a letter of M. As a result, if I was to encrypt the word of “CRYPTO”, it would be a word of “PQYNTM”

Thus we do this for the whole string of text that we want to encrypt and now

we have a string of text that we really can't read, but as long as we have that substitution cipher, as well as the decoder, we can go decrypt it. It is good to note that this is not a key; this is just simply a substitution cipher. We get into keys and the use of keys shortly.

Chapter 24 Viginere and Hebern

During the 16th Century, there was another cipher that was called the “Vigenere Cipher” The Vigenere Cipher was supposedly the first cipher that used an encryption key. Earlier when we were talking about the Caesar Cipher and I said you needed the key and I said that's not actually a key; that's just a shift cipher. Well, the key is what we have in the Vigenere Cipher.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

This is where the idea of using an encryption key comes from. If we were to look just briefly at the Vigenere Cipher, we have another example of that, and you can see in our example that it's a much larger table. It's similar to the shift that we saw in the Caesar Cipher, but we have it happening multiple times.

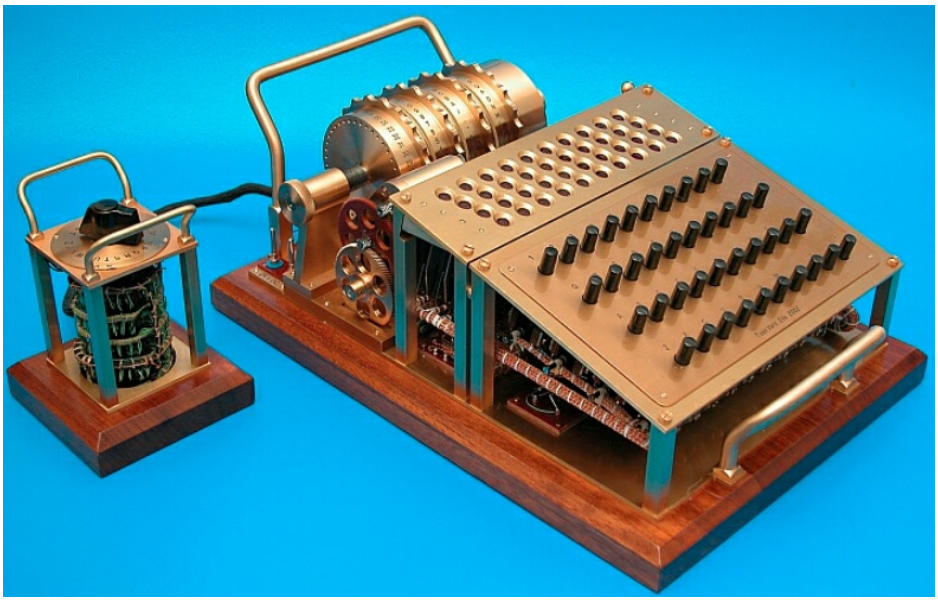
Along the left-hand side, you would see the key and then the cipher across the top. We're not going to go into a whole lot of detail on this, but just to give you an example in showing you the difference between going from a Caesar Cipher, which was very simple compare to the Vigenere Cipher. Well,

the Vigenere Cipher is more complicated. In terms of the strength of the cipher, this Vigenere Cipher is considered easy to crack these days, but back then, that was some cutting-edge technology.

Also at the start of the 19th Century, when everything started to become electric, and technology was advancing, there was E. Hebern who designed an electromechanical contraption. This was called the Hebern rotor machine. It used a single rotor that had a secret key embedded in the rotating disk. That key would encode the substitution table, and every time you would press a key from the keyboard, it would result in your cipher text.

Chapter 25 Enigma Cypher

Another popular machine is known as the Enigma Cypher machine. This was invented by a German engineer, Arthur Scherbius, at the end of World War I. The Germans used this machine during World War I. It used 3 or 4 rotor, so it was more complicated than the Hebern machine.



You would type, and the ciphertext would be the result. Poland eventually broke this, and then it was transferred to Britain, but this is a much more complex machine. You can see in this example that there were a couple of rotors there. You can see the four different slots there. It used three or four rotors; it depended on the machine. Some are used more rotors than that.

Fast forward into the '70s, a company called IBM formed the group called Crypto-Group. This was headed by Horst Feistel and he designed the Lucifer cipher. The Lucifer cipher is something that we all know today as DES or data encryption standard.

In 1973, NIST put out a request for a proposal for a need of a cipher. Lucifer was designed at this time, in fact Lucifer was the name that Feistel gave to a number of the ciphers that he had created, but at this time Lucifer was on the scene. NIST eventually accepted it and it became DES.

It started out as Lucifer and ended up becoming DES, or the data encryption standard as of 1997. DES had been broken by an exhaustive search attack, so this is bad news for DES, so NIST requested a proposal for a need for a new

block cipher and that's when work began, and people were trying to develop a new cipher that couldn't be broken like DES was.

DES eventually became triple DES, which was three DES processes: an encrypt, a decrypt, and an encrypt and so that made it a little stronger and it lasted a little bit longer, but DES by today's standards is very weak. In 2000, the NIST again accepted a cipher called the Rijndael cipher and that was named AES and it's called the Rijndael cipher.

In terms of the history of AES, what you would find is that the designers used their last names and they combined the two of those together and that's what they called the cipher. Once NIST adopted it, it was called AES or the Advanced Encryption Standard and that was brought in at 2001. It catches us up to today.

We've seen some of the different ciphers that were created along the way and how they got much more advanced. We'll take a look at some of the encryption algorithms a little bit later on and we'll discuss them briefly as we talk about how encryption works. Let’s move on and discuss hash algorithms and how they play a part in cryptography.

Chapter 26 Hashing & MD5 Checksum

Hash algorithms play a large part in cryptography today. They're designed to provide data integrity. What happens is we take an arbitrary length of data, whatever size that data is, we hash it, and the hash results in a fixed-length value. This fixed-length value that we have as our result is called a digest. That explains the term Message Digest 5 or MD5.

One of the algorithms that we use often. It's called a digest and it's similar to a fingerprint. It's unique, so we have the ability to use that unique fingerprint to verify the integrity of our data. When we do a hash on our data, there is a finite number of possible outputs to our algorithms. What could happen in some cases is that we put two inputs in, and that could result in the same output because we can only have so many hashes and the inputs can be variable lengths of data.

The term we use here, is a “hash collision” Hash collisions occur because of the finite number of possible outputs available to us. When do we use hashing? Well, hashing is used to verify the validity of an original message and it's also used in some other cases, for example, to verify software images. For example, Cisco uses these for every one of their software.

They give you an MD5 hash, so that when you download an ISO image or any software from them, you can compare it to the hash that they say it should work out to be. MD5 is not new and you can check validate software packages on multiple platforms, but for example on Windows 10, the validation requires the tool called “CertUtil”. In Windows 10 this utility built in to calculate the hash. Once you open your Windows command prompt and enter the following command:

“CertUtil -hashfile <path to file> MD5”

Depending on the size of the file it may take a few seconds to run the calculation, but if it’s successful, the MD5 hash will be displayed. It is also possible to generate checksums for other hash algorithms by replacing the MD5 parameter used above with either MD2, MD4, MD5, SHA1, SHA256, SHA384, SHA512. As a side note; if you don’t specify a value, then SHA1 is used by default. So, if all you need is to determine the checksum of a downloaded file, then there is no reason to install additional utility to do so.

Either way, once you have the checksum generated, you can compare it to the MD5 Checksum that for example, Cisco provides publically. If that hash generated to the hash that they have provided, it validates that the data is accurate.

We can also use this MD5 option to plug in a file name and run an MD5 hash on the file and then the resulting hash output of the file we can compare to an MD5 checksum that was sent along with a download of a file, and that's how Linux does it too with their ISO images. If you were to download an ISO image, they'll give you a hash, and you can validate the hash, just to make sure that you got it all and nothing changed in transit.

Chapter 27 Authenticating Crypto using Hashes

Often we provide authentication in a crypto world by using hashes. When we do this, we have to have two devices, and the two systems have to agree on a shared secret key. They have to do this ahead of time, but once they've decided on this shared secret key, they then use the key and a hash function such as MD5 to verify the data's integrity.

As we get a little more detailed into how these functions, we need to be aware of RFC 2104. RFC 2104 describes the key hashing for message authentication. When we configure routers we would configure MD5-hmac or sha1-hmac, and this just indicates that we're using the hashed message authentication code, which is using the hash function to provide authentication of our messages. Let's discuss how this process works.

Imagine we wanted to send some money to somebody. Let's assume we have host A and host B, and we want to send a check from host A over to host B, and we're going to send a check for $100. What we need to do ahead of time is that they need to agree on a shared secret key.

One way to do this is just to call each other and tell each other what each of their key is. Let's say they do that. They have their key, and that key has been agreed upon on both sides. They are also going to use a hashing algorithm and that hashing algorithm has to be decided upon.

They have to be using the same hash or they are going to end up with a different output. Let's say that the hash that they decide to use is MD5 and as long as they're both using MD5 on each side, then everything is going to be okay. Then they'll go ahead and take the data, or that check, and host A is going to take that check for $100, and take a copy of that check, and run it through the hash algorithm.

That hash comes out with a authenticated fingerprint. This is what we call the HMAC, which is a hashed message authentication code. Host A then going to pop that HMAC onto the check and send it along with the data. It then comes across to the other side to host B. Then on the other side at host B, he needs to take that HMAC that's on the data that was sent along with it, and need to set that HMAC aside. Host B then need to take a copy of that check, run it through the hash with that key. But, remember that the key ingredient is the key, and then we come out with that HMAC, and we compare that HMAC to

the HMAC that was sent, and this is how we provide that authentication.

It's not a new process, but it is an essential process for you to understand because this is how hashes are used in crypto to provide authentication. It's not the only thing that uses hashes. Crypto is not the only way that hashes are used for authentication. We also use this functionality for routing data authentication.

Imagine that we're sending a routing update to a neighbor, or in plain English, a router once would learn about a new route, that router would advertise that new route to all its neighbors by sending a routing update. Here, want to make sure that the routing updates that come from a neighbor are not modified in transit.

We don't want anybody injecting routes into our routing updates or messing around with our routing tables, so we put a hash on the routing update, and we compare on the other side or the neighboring routers. We also use this in IPSec gateways on firewalls, such as Cisco ASAs and other Cisco IOS, so that we can authenticate the encrypted data that's coming in.

Another place that this is also used is in TACACS, which is used for AAA authentication or AAA, which is Authentication, Authorization, and Accounting. TACACS stands for Terminal Access Controller Access Control System. TACACS sets up a session, and then the data in that session. We can hash those messages to make sure that our data is not being modified in transit. Crypto authentication is often done with hashes. It's not the only place we see it, but critical functionality of hashing today.

Chapter 28 Pros & Cons of Hash Algorithms

There are a few hash algorithms that we see in production networks today. For example, we see MD5, which stands for Message Digest 5. MD5 is a 128-bit digest, and while it's pretty quick, it's not recommended today because it's outdated. Therefore it's considered obsolete.

SHA-1 which is the Secure Hash Algorithm, it’s a 160-bit digest. SHA is much slower than MD5, because that 128-bit digest, but it was considered more reliable for some time. Nowadays, it's vulnerable to collision attacks so we want to avoid that as well. We rather want to use something like a SHA-2.

SHA-2 was adopted by the U.S. Government. SHA-2 provides different bit strengths of bit digests. SHA-2has sizes from 224, 256, 384, and 512-bit. One of the common ones that we see nowadays is the 384-bit SHA-2, and that's much more common these days. That's considered more secure so if you're going to choose one of these three, and you have SHA-2 available to you, I would recommend using a SHA-2, and that's probably what you're going to see more of, especially in Next-Generation Firewalls such as Cisco Firepower these days.

That does it for hash algorithms. Let's go ahead and discuss what encryption is, how encryption works, and we're going to see and compare asymmetric encryption to symmetric encryption. We'll discuss encryption keys as well as how we handle some of the issues that are involved in our keys.

Chapter 29 Encryption Basics

How does encryption work? Well, encryption is the process of taking a message and hiding the original context. When we do an encryption, we take the plain text data, and we convert it to a cipher text. That cipher text is supposed to be unreadable. Decryption reverses that process. It takes a cipher text and it runs it through a decryption algorithm, or the same encryption algorithm with shared secret keys.

Using those keys, it comes out with the clear text or the original message. We saw some of those older ciphers that were available to us. We saw the Caesar Cipher, and we talked about the Vigenere Cipher. Those are very weak by today's terms. Today we're going to use something like a DES, a triple DES or an AES encryption.

I would not recommend using DES anymore, although mentioning it just to have good coverage of the different protocols, but let's talk about how the process works and we're going to do this from the perspective of Cisco IOS. In Cisco IOS we have encryption functionality. That one of the software features in our Cisco router is that it can provide encryption.

Sometimes its encryption is done in software, while at other times, the encryption is done in hardware. When we do it in hardware, the process is much faster, so if we have a lot of data that we're trying to encrypt, we're probably going to want to do that in hardware.

Sometimes this is a licensed feature in Cisco IOS, but to be able to encrypt data, we have to have a key. Keys are the most important element of the encryption process because they absolutely have to be secured. We have to have the same key once we are doing a symmetric encryption. We need to have the same key on both sides. They need to agree upon that, or exchange that key before encrypting data.

What's going to happen is we're going to take that plain text data that's coming in, so our bits are coming in plain text. We could call that clear text, or we could just abbreviate it CT. In a lot of examples, clear text mentioned as only CT or plain text. They all mean the same thing. I don't want you to get confused there. So CT comes into the router. The router takes that key and the algorithm that we've chosen to use, and the CT, and it spits out the encrypted data. That encrypted data is a ciphertext. Therefore it is an

unreadable ciphertext that we should now be able to pass across the public network and not worry so much about that packet being grabbed or seen by someone because it is a ciphertext; it's unreadable.

As it gets to the other side, the process has to be reversed. The key that is used on the other side is now the decryption key, and it is used with the decryption algorithm, which is the same algorithm we used on the other side. Then it should produce the original message or the original data, and that's the process of encryption with a Cisco IOS scenario.

The process is also similar on other devices too, it's not that big of a deal, but just to give you an idea, we kind of put this in terms of a Cisco device. What's the danger of cryptography?

Well, let's think back in the old days. Back in the old days you had someone who wanted, in wartime wanted to send a message, maybe one general to another, so they would use one of those ciphers such as the Vigenere Cipher, and they would encrypt the data, and they would send it with a currier.

If that currier would be intercepted and the ciphertext would now be stolen, the data that is a ciphertext is in the hands of the enemy. What does the enemy do? Well, the enemy tries to crack the code. That's where cryptanalysis comes in, and that's a field of cryptography today.

There is a specific field called cryptanalysis, so we want to discuss what is cryptanalysis. We'll define it formally and we'll talk about some of the common attacks that we have included in the process of cryptanalysis.

Chapter 30 Breaking Cipher Text

Cryptanalysis is the study of, and the practice of breaking cipher text with the goal of obtaining plain text data. There are a number of common attacks. We're going to touch on some of those just to give you an idea, but the first attack I want to mention is a brute force attack. Every single encryption algorithm is susceptible to brute force attacks.

Brute force just means we're going to try every single possible key until we figure out which one it is. The longer the key, the longer it takes for an attacker to try all the possibilities, and that's why they say that it might take 1.4 trillion years for us to decrypt certain messages. That's, using current methods of brute force attacks, but that's just one type of attack and it is an attack that all our algorithms are susceptible to.

The next type of attack is a ciphertext-only attack. Ciphertext-only attacks is where you have the cipher text, and you're going to use whatever means possible to figure out what the clear text is. The only text you have is cipher text, so that's all you can use.

Maybe you captured it in transit. You've got the cipher text. Just like in the example where we had the currier that was taking an encrypted message, and he was intercepted, and now the data is in the wrong hands. That's the idea here. You've got it, and you're the wrong hands in this case, but you've got this data, and you're going to do whatever you can to analyze this and determine what the clear text data is.

Another type of attack is the chosen-ciphertext attack. In this case, you choose what cipher is decrypted, so you take the data, you get that cipher data decrypted, so you have means to get it decrypted, and then you see the plain text, and then you can look at the key space to see which key encrypted it.

The key space means all available keys within the space for the size key that you're using, so it means that all of the available keys can be used. That could be a bad attack if the attacker has time to do that.

Another type of attack that we have is called the known-plaintext attack. The known-plaintext attack is where you have some ciphertext, but you also know something about what's encrypted, and using the two of those, you've got a better shot at cracking that code.

The next attack I want to share with you is called the chosen-plaintext attack. In this case, you choose what data gets encrypted, so you have the cleartext. You get it encrypted, and then you look at the output, you observe the output, and try to figure out what's what or how it was encrypted, and how to crack the key from there.

We also have an attack called the birthday attack. That is a form of brute force attack, specifically on the hash function. It uses the mathematics behind what's called the birthday problem in probability theory, and that has to do with if you put 23 people in the same room, at the same time, then it's a 50% chance that two of them will have the same birthday. The last attack I want to mention is the meet-in-the-middle attack. The meet-in-the-middle attack is also known as or is a known plain-text attack.

Most of these attacks are not an exhaustive list, instead just giving you an idea of some frequent attacks. All these attacks fall under the area of cryptanalysis. Remember that Cryptanalysis is the study or practice of breaking ciphertext to get to the plain text data. That's the danger that we have in sending traffic across because there is that whole field of cryptanalysis.

That's why I mentioned earlier that the keys are the most important thing to keep secure and make sure nobody gets the keys because then we're giving away our data. Next, we want to talk about encryption and those keys, how they're used together, what details are involved in it. So we're going to look at symmetric encryption algorithms, and we're also going to look at asymmetric encryption algorithms next.

Chapter 31 Encryption and Keys

In the world of encryption and keys, we have what's known as a symmetric encryption algorithm. Symmetric encryption algorithms are going to use the same key on both sides, and typically these keys are going to range from 40 to 256 bits in length, but these days we don't want to use anything less than an 80-bit key. Those are considered weak by today's standards, so we want to avoid those if we can.

Symmetric encryption algorithms are fast, and that's why we use these for VPN connectivity. The one drawback is that key management can be a challenge. We have to have a standard secret key before exchanging our data, and we have to keep that key a secret.

What are the types of encryptions that are symmetric? Well, some of the common ones are DES, for example. DES is 56-bit encryption. It is weak by today's standards, but DES is a symmetric encryption algorithm. We also have 3DES, that's 168-bit encryption, but it's three DES processes, so 56, 56, 56.

We have AES, the Advanced Encryption Standard. AES comes in a couple of different lengths, 256, 192, and so on. I won't list them all, but AES is generally the accepted standard today. We also have IDEA, RC4, and Blowfish. Those are just a couple other symmetric encryption algorithms that are available to us today, and those are all symmetric, so once again they are all using the same key in sending and receiving, encrypting and decrypting.

We also have asymmetric encryption algorithm. With an asymmetric encryption algorithm, we use key pairs. With the key pairs, we have a public pair and a private pair. We generate both; public and private key pairs, and we never give out our private key; we always hold on to that.

If we encrypt the data with the private key, then we decrypt the data with the public key. If we encrypt the data with our public key, then we decrypt the data with the private key, so it's vice versa. This is a lot slower process then symmetric encryption, so we generally don't do this on real-time data.

That's why we don't use it in the VPN traffic that we encrypt, although we might use it for encrypting, or doing keys and use it in the generation of our keys. There are other asymmetric encryption algorithms too, such as RSA, DSA, or elliptical curve, and those are different algorithms that we can also

as an asymmetric encryption algorithms.

We have key lengths from 512 to 4096, so we get extensive keys with those asymmetric encryption algorithms, and they all have their place. Generally, we use symmetric encryption for the real-time data, while asymmetric encryption is not necessarily used for the real-time data. Let's discuss how both of these functions.

Symmetric encryption

We'll first look at a symmetric algorithm, and then we'll look at an asymmetric algorithm. With the symmetric encryption algorithm, we have our clear text data that comes in. We encrypt that data, using a key. We pass that data across, and now it's called encrypted data or ciphertext. We decrypt on the other side, using that same key that we've negotiated; using a shared secret key that's negotiated ahead of time and DES, 3DES, AES, RC4, all these are examples of this type of encryption.

Asymmetric encryption

Asymmetric encryption works a little bit differently, and I will share with you two different options. In the first scenario, we'll have host A, and we'll have host B. We are going to send traffic from host A over to host B. If we're on the side of host A, we're going to generate a public and private key pair.

If we have a public and a private key pair, we're going to send that public key over here to host B, and host B is going to hold on to that public key. Now, host B has our public key from host A. Next, we'll have a public key for host A and private key for host A, and this is how we'll differentiate between these keys.

Now we can take some data, send it through host A. Host A can use our encryption using the private key, send that using the private key, send that traffic over to host B, and then host B can use the public key to decrypt it. Now think about this for just a moment. When we encrypt data with our private key, everyone that has that public key can decrypt it. So this is probably not a real good way for me to keep my data private. So what would be a better way?

Well, let's reverse this process. Let's say that we have host A and host B and

we've already gone through the process and host B has the public key of host A. Host B can go ahead and encrypt data using host A's public key, send that data over to host A, and then host A can use the private key that nobody else has to decrypt that data.

By doing that, we now have secure ciphertext or encrypted data traveling across the network that nobody, except for host A is going to be able to decrypt. That's how the asymmetric algorithms work. What you might be thinking now is well, that's great, but you are concerned about these keys.

The keys tend to be a problem, exchanging the keys and making sure those keys are trusted, we're going to get to that shortly, but it's essential to understand the process. For now, we have discussed the process of symmetric algorithms and asymmetric algorithms and how they work.

Once again, the symmetric algorithms using a single key and the asymmetric algorithms using a public/private key pair. Let's move on and talk about digital signatures because it relates to the asymmetric method that we just discussed.

Chapter 32 Digital Signatures

If you remember when we encrypt data with a private key, everyone with a public key can decrypt it. That's not a great idea. Digital signatures are giving us a way that we can send a piece of data, such as a transaction, or email to a customer and have them verify that we've signed it, as it’s been agreed by us that it's not forged.

Once we use asymmetric encryption, we could take the data, encrypt it, send it to the other side, and if they have the public key, they can decrypt it. That's a good start so that we could distribute our public key to a bunch of people, and then they could use that public key to ensure that we're the only ones that are sending them data. This is because we're encrypting it with our private key, and they're decrypting with the public key.

They can verify it, and they can authenticate that it's us; because they can read our digital signature. But the data that we send is variable lengths, and it's significant in some cases, so computationally, this could take time. We could consider this to be expensive too.

So instead, what digital signatures do, is they take the data, then hash the data, and then encrypt the hash, because the hashes that we send are a fixed length. Then on the other side, when they get this encrypted hash, now they have this encrypted hash that is a fixed length and it’s easy to decrypt.

It's not computationally expensive to do this, so it's much more efficient. That is an idea of how digital signatures work, and we can use that data to authenticate something that we've sent, or verify our digital signature. We did talk about how things get complicated in terms of public keys being distributed, and we run into the problem of, how do we know that I have the real public key of somebody.

How do I trust that it's their public key that I got? The answer is PKI. PKI is where this whole thing comes in to play. It helps us to ensure that we have the trusted distribution of our keys. We could think of this in terms of the Department of Automobiles. In the Department of Automobiles, we go down to the DMV and we apply for a driver's license. We provide them information, and they verify the information, they issue us an identification card, an ID, a driver's license. Now, if we're driving and get stopped by a police officer, that police officer wants us to identify ourselves.

He's authenticating who we are. We show him our driver's license. That driver's license was issued by a trusted third party. When he looks at that driver's license, he's verifying that it is us, that it's signed by the trusted third party, that it's within the validity periods.

Because drivers' licenses expire; so do digital certificates which are what we're leading to. He verifies that it's not within the expired time or it's within a valid date and then he might radio in to make sure that it's not suspended. All these are things that in real life we do, we see these all the time; they're commonplace to us, but they're also used in a PKI environment.

Let's look at how PKI works. It's a pretty simple process. It's just there are a lot of pieces involved and because of all the pieces that are involved, sometimes people get a little bit lost, but there are some essential things that we need to understand.

First of all, we have a CA-s or certificate authority. That certificate authority is a trusted third party. We have different CAs available to us, such as VeriSign, Entrust, Microsoft Windows certificate services, Cisco IOS, in fact even Cisco ISE, identity services engine can act as a CA server. But CA-s are our trusted third party.

You can look at them as being the Department of Automobiles. Imagine the following scenario. Me and my friend, whose name is Bob, we want to talk to each other, but first, before we try talking to each other and identifying ourselves to each other, we're going to register with a trusted third party.

I'm going to submit a certificate signing request aka CSR to the CA server, and I'm going to say that this is all the information about me. Here's my public key; so give me something that identifies who I am. So the CA will send back to me an identity certificate. Now I have this identity certificate.

Remember, I sent the CA my public key so he validated, and digitally signed that certificate which has my public key, and I'm going to also retrieve that from CA server. I'm going to retrieve a root certificate, and that root certificate is the CA's public key. So, I download the root certificate and that's going to let me read the signature that's on an ID certificate that the CA has signed for me. Now my friend Bob is going to go through that very same process. Bob is going to send his certificate signing request over to the CA server along with his public key.

He's going to tell the CA about his public key, and will provide all the information, so he can digitally sign it and identify who he is. The CA is going to give back his identity certification, and he's going to digitally sign that identity certificate.

Now Bob is also going to download the root certificate so that he can read signatures that are signed by that root. This is where it becomes interesting. We've both enrolled with a trusted third party. We both have our own identification card that says who we are, and until when our certificates are valid. This is when they expire and so on.

This is the location of the list that tells whether it's suspended or not or revoked. Now we have these identity certificates, similarly to a driver's license that have a nice picture us. So now I want to talk to Bob. So I say, hi Bob, trust me. Here's my identity.

Now Bob is going to take a look at that identity. He's going to look at that signature on the bottom of that identity certificate. That is a digitally signed signature. Remember, we just talked about how something is digitally signed.

So he looks at that digital signature. The only way he can read that digital signature is by using the root certificate because it has the public key of the root, so he looks at that signature, uses that public key to decrypt it, to verify that it is accurate. Therefore he is going to trust me that's who I am.

He now has my public key, so he will send me his digital signature. I go ahead and do the same process. I look at that signature that's on his ID. I use the root certificate that I downloaded from the CA server to validate it.

Once I validated it, then I go ahead and trust his identity certificate, and now we can talk. We both have each other’s' public key, and we can use that for encryption if we'd like. We can use that for authentication if we'd like, but that's the process of PKI. It's a pretty high level, but we will dig a little bit down into the process of how the root certificates or CSRs work and how the ID certs are all used.

In summary, so far, we did an excellent overview of cryptography, and we looked at the history of cryptography. We saw the Caesar Cipher, we saw the Vigenere Cipher, we talked about the Enigma machine, we talked about some of the newer encryptions that we have as well.

We also talked about hashing and how important that function is to us in cryptography. We looked at encryption methods and covered the difference between symmetric and asymmetric encryption. We also talked about that other field that relates to cryptography, called cryptanalysis and how if that's done correctly.

Then it's an excellent way to test our cryptography algorithms to make sure that they're secure, but if it's done in the wrong way, then it is a way to obtain the clear text data, basically cracking our crypto, so that could be bad. We also talked in terms of cryptanalysis on how the key is essential to us, and so that's why we discussed keys at length.

We also talked about digital signatures and PKI environments. That's a decent look at what we have talked about in the last few chapters. Next, we're going to talk about network topologies, and look at some devices that have specific security functions. We're also going to take a look at general secure network architectures.

Chapter 33 Network Topologies & Firewalls

We're going to get into a lot of details on how firewalls function, but why are we bringing these up in terms of network topologies? Well, this is because firewalls are a huge part of secure network topology. You can't have a reliable network topology with some point of access control between your networks and in general, so that's where firewalls come into play. What do firewalls do? Well, firewalls provide access control between different zones.

The zones that we're talking about could be something like an inside zone and an outside zone, or an outside zone and a DMZ. Firewalls are providing specific access control between zones. It could be in the form of access control lists, or it could be in the form of a modular policy framework with advanced protections filtering such as HTTP or FTP.

If you're a firewall, you should be a hardened operating system. We don't want an operating system that is vulnerable if it's going to be protecting our network as a system. You have to think about your entire architecture that needs protection. You've got your routers, switches, servers, end-users, load balancers, whatever the case may be, they all require protection.

If you have a firewall sitting on the edge of a network, it needs to be a reliable device that is going to be resistant to attacks. It needs to be a hardened operating system. It also need to operate inline. Your packets need to flow through the firewall, and the firewall can't filter the packets. Usually, there are several other features that go along with firewalls.

For example, with the Cisco ASA or Adaptive Security Appliance, not only does it act as a firewall, but it has routing functionality, and it also has VPN functionality. It does have intrusion prevention functionality too. Let's look at a scenario of how a firewall would operate.

Imagine that we've got our PC, we've got an ASA, and we've got a server. Let's imagine that the PC is on the outside, and he's trying to make a connection into the server, and let's just say that the server is an FTP server. We know that FTP as a protocol is using TCP port 21, therefore that is the destination port. The source port is going to be some random number.

So that ASA sitting there on the perimeter of the network is going to have to make a decision, if it should allow TCP port 21 traffic coming into the network or filter this traffic. As that traffic comes in, the firewall will do a

couple of things. It's going to look to see if it has access rules.

It might take a look at an access control list that's applied on the outside interface. It might take a look at the modular policy framework configuration, or MPF and look specifically to see if there are any inspect configurations that dictate what should happen with this traffic.

We also may need to do some network address translation. So on the outside world, the PC might be connecting to our server from a public IP address. It could be something like a 205.30.25.3 and on the inside that real server's address might be something like a 192.168.1.10. If that's the case, then the ASA is likely going to be configured to do static NAT, and it'll have to provide the NAT translation as well as any access control that comes along with it.

There is a lot going on with a firewall in this simple scenario, but it is important to understand that in some cases, traffic is going to come in. Let's imagine that somebody else on the outside is trying to send in some traffic on port 2345, and it's a TCP traffic and it's inbound. The firewall decides if that traffic is need to be blocked or not.

If it’d not traffic that the ASA wants to see on the network, it will discard it. That's the firewall functionality. The device can say “no”, “this traffic is not allowed”. We'll get into more details on how firewalls function, but often we're going to see these operate at the edge of a network.

Yet, that's not the only edge that we have in our network topologies. We also have edge of the data center. It could be the internet edge or the data center edge where we see a firewall implemented. Those are the two common locations where we'll see this scenario. Let's move on to discuss intrusion prevention systems.

Chapter 34 Intrusion Prevention System

IPS stands for Intrusion Prevention System, which is designed to do in-depth packet analysis. It's designed to look deeper into the packets than perhaps a firewall would look and analyze. There are many different types of IPS that we can perform if we were to do anomaly detection, but deep packet analysis is the key to intrusion prevention.

In the old days, our intrusion prevention devices used to be out of band. Thus traffic would flow from the outside to the inside of the network, and we would copy that traffic over to an intrusion detection system. The problem with that is that it we would always analyze traffic after the fact. The packet was already forwarded, but a copy of it was sent to the sensor. We could block subsequent packets, but the initial packet in an attack wouldn't be blocked.

Today's intrusion prevention systems operate inline. This way, they can look at traffic as it comes through, and before forwarding that traffic, they can determine whether or not it needs to be discarded, whether it needs to be forwarded, or if we need to do any normalization on the packet.

Normalization means we're modifying something in the packet. For example, if we were passing TCP traffic through the firewall and there was a particular TCP option that we don't recognize, or we don't want to see on the network, we could clear out that option before we forward the packet. In addition to that, it's essential to understand that an IPS does not replace a firewall.

Usually, an IPS is going to be slower than a firewall, so we don't want to send necessarily all traffic to the IPS. Perhaps we only send essential traffic. We want to put IPS inline, in a place where it's not going to end up being a bottleneck in the topology, but somewhere where it will be able to filter traffic that is going to be vital to us.

Let's go ahead and take a look at an overview of how an IPS would function. First of all, let's assume that we've got an IPS sensor, and it's in between two routers. We could put switches in between too, but the IPS has to be inline between traffic. As traffic passes through the IPS sensor, it's going to do deep packet inspection. There are several different things that we can do with an IPS, but for this scenario, let's just assume that the SNMP packet that's coming through the IPS sensor. The IPS sensor is going to look at that SNMP

and look at how the protocol is designed, so IPS knows how the SNMP protocol should be used, and it can check if the SNMP packet compliant to the protocol standard. As it analyzes the packet, it can say yes or no.

If the answer is yes, then it'll go ahead and forward the traffic. If the answer is no, then it will deny that packet. That's not the only thing we can do with an IPS, that's just one example. Another thing that we could do with an IPS sensor is that we could have several packets that are coming through the device.

Those packets could be fragmented, so if we could have fragment 1, fragment 2, fragment 3, and as those packets are coming through, the IPS can reassemble those fragments and then look at the data to see if it is compliant to the protocol standard. We can also look at the payload, and we can determine based on the payload if this is some form of a known attack.

Another thing that we should point out in terms of an IPS is that is signature-based, which means that we have to continually update it. The reason for that is because attacks change, so the signatures need to be updated to be prepared to handle that type of information. That's just another example of the functionality of an IPS sensor.

We'll use the devices on either side of the IPS sensor to determine what traffic gets routed through that sensor. In this case, we've got two routers so we could route traffic for a specific subnet through the IPS. If it's our server subnet, we could route that traffic through the IPS to ensure that it's being inspected.

There are other ways to route traffic. We could use VLANs, we could use VLAN groups and so on, but this should give you a good overview of how IPS functions. Let's go ahead and discuss content security.

Chapter 35 Content Security Basics

IronPort has been purchased by Cisco, which is a really solid content security portfolio that consists of the web security appliance and the email security appliance. If you want to learn more about IronPort, please visit the following link: https://www.cisco.com/c/en/us/about/corporate-strategy-office/acquisitions/ironport.html

The Cisco WSA and ESA are designed for specific types of applications. The ESA is designed for email traffic, while WSA is designed to filter web traffic and that's what content security is all about. These devices filter traffic of a specific type. Let's discuss how content security functions if we were using an ESA in our network.

First of all, imagine that we have a user on the outside of our network. They're going to be sending an email message to the inside of our network. As that traffic is passed in, the Firewall is going to forward that traffic over to the ESA. This is where the ESA provides its services. It's going to look at that traffic and it's going to try to determine if there is malware in that traffic.

If the email message does not contain malware, and there is no malware, then we're going to forward that packet on to the destination. If there is malware, if we were to look at it and the answer is yes, there is malware, well, then we need to clean that email before we send it. We're going to strip out the malware and only then send the email to its destination. That's what the ESA is going to do for us. It's going to ensure that the email traffic that we get is clean. We don't want any emails coming in and employees clicking on that, then from there infecting or having that traffic spread to the rest of the

network. If we put one of these devices in our network, it should be seated in a location where the email comes into the network. We have our Firewall on the edge of the network, and that Firewall can forward it over to the ESA before it making its way into our network.

You can isolate the device onto a DMZ, and that way, you keep that bad traffic separated. We don't have to worry about it compromising our internal network, and we end up with clean emails. That's an example of content security if we're using an ESA. Let's discuss what it would look like if we were providing content security services with a WSA.

In this scenario, imagine that we've got a user on the inside of our network. But, let's determine our zones first. The inside zone is on the right, the outside zone is on the left, and the DMZ is where the WSA is located. Next, we have our user's computer on the inside, and he's going to make an outbound request for a website.

Let's imagine that our user is going to visit Google.com. The firewall is going to have to get that user's traffic forwarded over to the WSA for inspection. We're going to look at the URL. We're doing URL filtering, and WSA starts looking at the website, and compare it to a list of trusted sites. He's looking to see if the website is trusted or untrusted.

In this case, it's going to be trusted, so the WSA will forward the request, which will then be sent out to the webserver. That web server can then reply, and the user can communicate freely. As long as it's on our trusted list, we'll go ahead and allow that traffic out. That's the first thing that happens.

Let's assume that we're going to go to someplace that's not trusted. In this case, the user is trying to go to a website that contains gambling traffic or something similar. In this case, the Firewall is going to get that traffic over to the WSA. This isn't the only way we could get traffic to the WSA, it's just for our example's sake.

The WSA is then going to determine if that is a trusted website or not. In this case it is a gambling site. It's on one of our blackout lists, so the WSA is going to send the user back the URL redirect, and that user is going to get a page that says, "access denied - you're not authorised to visit that website". The idea is to provide protection, so if a user is trying to access sites that are deemed inappropriate by corporate standards, and then we can go ahead and

prevent that.

There's a lot more that we can do with the WSA in terms of its capabilities. It does integrate with Cisco and the security information operation center, the SIO out online, so there is a lot of correlation and updates that go with the WSA. This gives us a decent idea of how content security is going to function when we're dealing with it in the context of users accessing websites.

Chapter 36 Remote Access VPN

We've already discussed cryptography, which is a significant part of a VPN connection, but VPNs are designed to carry private data over a public or a shared network. Usually, this is done at layer 3, so we're going to take our IP traffic, encrypt it, and tunnel that traffic. We could put a brand new IP header on, and that way we can route private IP addresses over a public network.

VPN is designed to provide confidentiality, integrity, and origin authentication. Let's look at how VPN works. We're going to look at it from two aspects. We're going to look at it from a remote access mode, and we're going to look at it from a site-to-site mode. The first scenario is the remote access VPN.

In a Cisco world, first, we're going to install the Cisco AnyConnect Client on our device, and that device can make a VPN connection generally into a Cisco ASA Firewall. The ASA is our primary VPN device in a Cisco environment, especially for remote access, but Cisco routers also capable of doing VPN connections.

When we're talking VPN with a remote access connection, AnyConnect can do SSL VPN or IPSec if we're using IKEv2. Important to differentiate between IKEv1 and IKEv2, because the AnyConnect Client does not do IKEv1 connections, which is a traditional IPSec connectivity.

There is an old Cisco IPSec VPN Client that was used, and we generally don't use that anymore. We mostly use AnyConnect Client, which at the time of writing this book is version 4.8. So, imagine that we have the AnyConnect Client installed on the mobile device. We point it to the address of our VPN gateway, which in this case, our Cisco ASA Firewall, then the user connects to it.

If this is an SSL or an IPSec session, we're going to call this full tunnel, and if this is a full tunnel connection, we're going to issue an IP address from a pool range on our ASA out to the client. The client will get a private address, and that address will be routable on the inside of our network.

Now that the user is connected, his traffic can be routed to our internal servers. The internal server sees him as the IP address from the address pool. Perhaps it's something like a 10.x.x.x or 192.x.x.x address base. When the

server responds to that private address, our network on the back end is going to forward that traffic towards the VPN gateway. The VPN gateway is going to get that traffic, and this traffic goes inside of a tunnel and it's going to shoot it back in that tunnel that we have established from the VPN Client. That's an important aspect.

The client needs to be able to get an address to be able to do the full tunnel type of connectivity. We can do filtering on the ASA to control what kind of packets the mobile device can push through. We're going to use ASDM in most cases to monitor and well, but really to configure our device. ASDM is the GUI version of the ASA.

In the ASDM, there is a client editor. It's a client profile editor and that editor allows you to make changes to the VPN Client's functionality and then that way, when the user connects, the user can negotiate to what VPN Client profile should it use. Based on what we've configured in that editor we can then push those changes back, and those would be applied.

We can also do split tunneling. Split tunneling means that the user will be connected through the VPN, and we can have it set up, so only traffic that is going to our internal networks is sent in the VPN. Everything else that's going to internet website such as Google will be routed through the local internet connection and out to the internet. This is just another way that we can control the traffic with a VPN. This is a remote access VPN. Let's take a look at how this would work in a site-to-site VPN.

Chapter 37 DMVPN & Site-to-site VPN

When it comes to site-to-site VPN type of a connection, we're talking about IPSec in which case our routers are going to form a tunnel between sites. We're going to generally have either a routing protocol direct traffic over the tunnel, or we're going to have what's called a crypto ACL.

That crypto ACL will define the local and the foreign networks in which traffic will be encrypted. We're going to use the routing functionality in the device to determine how we get from one side to the other. At this point, we have the ability to take all packets from an inside host, move them across the tunnel, and deliver them to a foreign destination, or a host on another network that's within our organization.

That's simple, but let's back up for just a second. That is just what we call a site-to-site VPN. Another type of VPN that's fairly common in Cisco world is what's called a dynamic multi-point VPN. in a dynamic multi-point VPN, we are going to have one device that will be known as the hub, and the other devices that are going to be known as a Spoke and each of those Spokes are able to establish a tunnel into the hub and pass traffic to the hub.

But they'll use a protocol called NHRP or Next Hop Resolution Protocol to be able to resolve the address of a Spoke device so that they can do Spoke to Spoke tunnels dynamically. Now you can see we have a full mesh type of network where I can go from point A through the tunnel to point B, or I can go to point A through another tunnel to another Spoke to destination C on the other end.

That Spoke to Spoke tunnel is dynamic, so when I don't have traffic between there, it can tear that tunnel down, which makes it an excellent way to establish my VPN connectivity. It's an excellent way to handle things. This is called DMVPN. DMVPN, dynamic multipoint VPN. It's a popular concept in the Cisco world, something that you'll run into as time goes by.

This should give you an overview of regular site-to-site VPN as well as the DMVPN capability that we have in our Cisco routers and that's another point that we should mention here. DMVPN does not work on Cisco ASAs. Part of the protocol specification or part of the DMVPN functionality requires that we use something called GRE, Generic Routing Encapsulation. Cisco ASA Firewalls do not support GRE, nor do they support NHRP. While they do

IPSec site-to-site tunnels, a Cisco ASA will not support DMVPN because it doesn't do NHRP, and it doesn't go GRE.

Chapter 38 Securing Endpoints

When we talk about endpoint security, we're talking about applications that run on a laptop or a desktop PC, so typically these are end-user devices. That would include something like a Windows firewall, antivirus or anti-spyware. When we talk about all these different devices, we talk about IPS, endpoint security, firewalls, or VPNs, and they all lend themselves to a secure architecture.

Once we talk about a secure architecture, we want to ensure that as we design our network and build out topologies, and connect all these devices and start implementing services on the network, we take a defense-in-depth approach. A defense-in-depth approach, start with a firewall at the edge, followed by an IPS.

As we get closer to the core devices or to the users, we're going to have authentication, authorization, and accounting, what we call AAA. AAA handles any validation, if we know who they are, whether they should be there or not, and then account for what they do. Beyond AAA, we need to have hardened devices, as well as our endpoints, need to be hardened too.

The way that we achieve that on our endpoint devices is with antivirus, anti-spyware, client-side software, or anti-malware devices. Another design recommendation, as you think about building a network is that you implement a least-privilege model. You only give people enough to get their work done; that's all the privilege they need, and then you increase it as it needs to be.

It's also recommended that you assess where the weakest link is. Your network is only going to be as strong as the weakest link, so if you step back and assess the network, as you're designing it, look for that weak link and then make the modifications there that are going to build an overall more reliable network.

Some things that we want to keep in mind in terms of design and the last thing that we want to cover here are some common network architectures. These are some terminology that you need to understand. The first architecture is called the Campus Area Network or CAN. CAN is a network that is in a campus area, perhaps between a couple of different buildings.

Here, we have network switches, and maybe a few routers. That's generally

where we're going to see our users at. We also have Cloud and Wide Area Networks or WAN. Wide Area Networks connect campus networks between a large geographic area. Older terminologies used to refer frame relay connections or ATM connections too. Another typical architecture is Data Center, and lastly, the Small Office Home Office, also known as SOHO Network.

Chapter 39 Managing Network Devices

Let’s begin with the router. The router can be broken down into multiple planes or areas of responsibility, including the data plane, the control plane, and the management plane. The management plane is responsible for handling configuration, CLI commands, and the GUI interface.

The control plane is responsible for building routing tables and providing forwarding tables to the data plane. We have protocols such as SSH, handled by the management plane, routing protocols such as OSPF or EIGRP, or BGP handled by the control plane, and the data plane is responsible for moving packets between interfaces.

Having this basic understanding, you know that if the management plane is compromised, the configuration can be modified, control plane functions can be impacted, and ultimately, the data plane may not be able to forward packets. Primarily, we're talking about the real possibility of your router becoming unusable.

It could even become an attack vector leading to other types of attacks on the network, like routing protocol corruption or man-in-the-middle attacks. We must protect the management plane. Here are some recommended management plane practices. Recommended practices for the management plane include password policy, role-based access control, authentication, authorization, and accounting or AAA, Network Time Protocol or NTP, the Simple Network Management Protocol version 3, or SNMPv3, and restricting access to management protocols.

Let's discuss first password policy. Routers and switches don't have default passwords configured. This is one of the first things we're going to want to do when we stand up a box, and there are multiple places where passwords should be enabled. We need to enable password authentication for the console, for access to Enable or Privilege mode, and for the VTYs, where Telnet or SSH connections terminate.

In addition to setting passwords, it's recommended that we enable RBAC. RBAC stands for Role-Based Access Control, and it's a means for us to define various user roles and have specific commands tied into each role. Roles can be nested, giving us a modular type of configuration syntax. The next recommended practice for securing the management plane is to

implement AAA. AAA can be done locally, or more commonly used with an external server such as Cisco's Identity Services Engine or Cisco ISE.

These external services integrate with Active Directory and allow you to use your existing database to authenticate users, authorize their activities, such as the commands they enter, and account for everything that happens on the device at CLI.

We also want to be sure to set accurate time on our network devices. It's easiest to use the Network Time Protocol, or NTP to do this. Using NTP allows us to synchronize time among all the devices in the organization. This way our log timestamps are accurate and any certificates that may be in use can adequately be checked to see if they are within their validity period.

Another form of remote management is SNMP, or the Simple Network Management Protocol. SNMP comes in three different versions, version 1 and 2 using clear text community names for authentication and not providing any confidentiality. It's recommended that you use SNMPv3 in today's network environments.

This lets us make use of the AuthPriv capability, and use stronger authentication and encryption techniques. Finally, we want to be sure to restrict access to management protocols. Telnet is a standard protocol used for remote management. However, Telnet doesn't use encryption, and therefore it's not secure.

A man-in-the-middle attack could capture Telnet packets and view the commands in cleartext. This includes passwords entered on network devices. For this reason, you should restrict the use of Telnet and use SSH instead. SSH is secure because it uses encryption and hashing for integrity checking. SSH version 2 is the current version that should be preferred. You should also use Access Control Lists to limit the addresses that are authorized for management. Next, we'll take a look at AAA concepts.

Chapter 40 AAA

AAA, which stands for authentication, authorization, and accounting. Fundamentally, who are you, what are you allowed to do, and then when you did it, we keep the logs of that. It's a straightforward process, it's something that we've had around for a long time, but “tripe AAA” is how we call it for short, and we understand that could be any of those processes at any given time, authentication, authorization or accounting.

You have to authenticate a user before you can authorize them. There are two types of AAA. We have AAA for administrative access and we have AAA for remote user access. In addition to these two types of AAA, when we implement AAA for either administrative access or our remote user access, we have four ways that we can apply it.

There is the self-contained method, which means we're going to do all of our configurations on our device, were we are going to create local username and a password in a local database. We're going to configure our AAA commands to check that local database.

We can also use the Cisco ISE for Windows. We install the software on Windows, it installs as a service, and it runs as a server, then we log into the GUI interface and we configure it.

We also have Cisco Secure ACS appliance, which is a purpose-built hardware appliance that runs ACS service software. The same interfaces as what we would see for Windows, maybe a little different in the graphic interface, but same concept.

Then we have Cisco's Identity Service Engine or Cisco ISE, which is a newer product, in fact it's newer than Cisco Secure ACS. Cisco ISE is built for user authentication, it's primarily used in TrustSec deployments with BYOD and secure wireless networks, but it branches to other parts of the network in doing authentication.

Chapter 41 ACS & ISE

When it comes to Cisco ISE and ACS, I want to give you an overview of both devices, so you have a bit of an understanding on the differences between the two of them. When we talk about ISE and ACS, the first thing to understand is that they're kind of very similar to each other. For example, if you look at Cisco ISE, a lot of its capability was ported over from ACS because ACS came first.

With that said, let's discuss some of the features of ACS. When you talk about Cisco TrustSec, for a long time, ACS was the core policy component. It is a Linux-based appliance in most cases. There used to a Windows-based that was popular too, but the hardware appliance seems to be more popular nowadays.

It does have a software operating system image that you can run in VMware. Once it’s installed, the underlying operating system, the command line configuration, is similar to Cisco ISE because it runs the application development image, and ACS runs as an application on top of the ADE OS, and that's the same with Cisco ISE.

The Cisco ISE application runs on top of ADE. So both ACS and ISE can be accessed via a web-based GUI for most of the configurations that you can do. ACS is an attribute-driven, rule-based policy model, but ISE packs a lot more features. ISE has in the functionality that exists in ACS, and up until version 2.0, it did not have TACACS+, so that's where ACS was still necessary.

ACS runs TACACS+, as well as Radius. ISE, up until version 2. 0 only did Radius. In 2.x versions, it does TACACS+ as well as Radius. ISE integrates the ACS functionality and the NAC solutions that we had in the Cisco NAC Appliance. ISE can do access control, and it can do profiling, it can do posture assessment to see what the status of an endpoint looks like, to see if it is running the certain antivirus that we expect to see, and if not, it can put it into remediation.

ISE handles guest lifecycle, so it can onboard a guest and provide them with a username and password. ISE also available as an appliance or a virtual machine. The virtual machine is excellent because it comes as an OVA, and we can deploy an OVF template and drop it right into VMware. That's just a little bit of a comparison between ISE and ACS, but they're very similar to

each other. As far as the protocols TACACS+ and RADIUS go, it's crucial that you have some understanding of them. The protocols themselves, they are a communication protocol so that our network devices can talk to our server.

That server is going to be ACS or ISE. ACS supports both TACACS+ and RADIUS, and ISE supports both as of version 2.x, but before that, it was just RADIUS. TACACS+ uses TCP port 49. It is considered to be more secure than RADIUS because it encrypts the entire packet, not only the password, like RADIUS.

RADIUS does have a more robust API and accounting capability than TACACS+. RADIUS is a UDP-based protocol that uses ports 1812 and 1813. The planned replacement for RADIUS is called DIAMETER. We use RADIUS for remote access, such as dial-up connections. We also use RADIUS for 802.1X connection.

802.1X connections would be for connectivity to a wired or a wireless network. We would use a supplicant, which is the client that's sitting on a native device. It could be the native supplicant, but right into Windows or Mac or Linux, and it'll authenticate back to our ISE in most cases, or ACS, depending on how we've built our TrustSec deployment.

We can also use RADIUS for SIP authentications. TACACS+ is not compatible with some of the predecessors, TACACS and XTACACS are a Cisco proprietary protocol, and what's unique about it, is that it separates the authentication and authorization process into separate connections. It does a three-way handshake, so it does authentication, then tears that handshake down.

RADIUS on the other hand, does authentication and authorization at the same time, in the same session. TACACS+ does support a large number of features, and primarily, it is perfect for administrative authentication and authorization. That's just an overview of TACACS+ and RADIUS.

Let’s move on and discuss SSH briefly. The protocol called SSH, is also known as Secure Shell. SSH allows us to create a management connection that is secure and encrypted. On Cisco devices, that SSH daemon that is running, accepts incoming connections, it works with both our commercially available SSH clients like Secure CRT, as well as some of those free ones such as Putty.

It doesn't make a difference in what we use. If you're running a Mac, then right from the terminal interface, you can create an SSH session. There are two versions of SSH, version 1 and version 2. SSH version 1 is only supported within older IOS environments, and version 2 is only used in a newer versions, so you're going to want to ensure that you look at your IOS versions if you're planning on running SSH version 2. SSH version 2 is the recommended version.

Chapter 42 Privilege Levels

Privilege levels on Cisco devices allow us to control the access that administrative users have to the commands that they can enter. When it comes to privilege levels, there are two privilege levels that we can work with.

We can work with level 1 and level 15, and then we have a number of them in the middle. We can configure different privilege levels, and that'll let us do some role-based access control. We can set different passwords that control who has access to various privilege levels and then within each privilege level, and we can define the commands that are available at that level.

There are 16 privilege levels in total, 0-15. Level 0 is reserved for the user level access privileges, and levels 1 through 14 are levels that we can be customized. Level 15 is reserved for privileged mode, or what we might call enable mode, and if we want to assign privileges to any of those other levels, maybe 2 through 14, we're going to use the privilege command.

Whenever we login to user EXEC mode, we're automatically put into level 1, and then as we enter commands, they're all executed at level 1. If we wanted to execute commands that are above that privilege level, we type the command “enable”, and that puts us into privilege EXEC mode, which is privilege level 15.

We can control which privilege level we go to once we've created additional levels. Under the default configuration, you end up in user EXEC mode, which is level 1, and then from there, you don't have a whole lot of access to anything, but once you get into enable mode, you have access to everything, and that's by default.

Regards to Parser Views, once they are configured, they give us even more control. The role-based CLI feature allows us to create different views of device configuration, and those views can be accessed by users that are at varying levels of visibility. For example, you could create a user called a sysadmin. That sysadmin might only be able to look at interface details or statistics, maybe not even looking at the configuration.

We can limit what they have access to. When we create a Parser View, we're going to define the commands that are accepted inside that view from a user, and what is going to be visible to them. It gives me better control using a

Parser View than I would if I were just to create privilege level commands for every single command. So, privilege levels are one way, Parser Views is another way.

To create a parser view, we're going into enable mode and then enter global configuration mode, and then from there, we'll go ahead and create a parser view. Once we've created that parser view, we'll set the views password, and then we'll go ahead and define what commands are available in that particular view.

Once we've created that view, we can then access the parser view. Before we create this parser view, you should note that the AAA new-model command has to be enabled. To access the parser view, we have to type the “enable” command, and then the view name. This will prompt us for the password that's assigned to that view, and then inside that view, we can type the command “show parser-view all,” and it'll show us all of the available commands we have inside that view.

Chapter 43 Syslog & Reporting

When it comes to secure management and reporting, we have some considerations that we're going to look at first. There are also some recommendations that we want to apply to our networks. The first thing that we want to ask ourselves when it comes to logging is what is it that I should be logging, what kind of traffic needs to be logged, what type of activity requires to be logged and what messages are important to us?

Once we figure that out, Cisco can offer an easy way to get the log messages and send them to an external server. But the standard protocol that is used for this purpose is Syslog. Syslog is a UDP-based protocol, and then the next question that comes up is what if our traffic is tampered with, and so what can happen there?

Well, this opens up new issues, if our logging messages are tampered with, because in some cases these log messages are required for legal matters, and if they're tampered with, then the problem that we have is that they can't be used in many situations. There are certain things that we need to ensure.

We need to make sure that we have the right timestamp on our devices, or on these log messages that are coming into the Syslog server. Also need to make sure that when something says it happened, that's when it happened.

Not only does this help me when it comes to security-related incidents, but also in troubleshooting different things it's good to know exactly when it happened and build a timeline, and we can see if something is being correlated over some time, or how much more critical those timestamps are going to be.

What do we need to keep all these under legal limits? Well, you should only keep the information that's going to be relevant to you, and store that persistently, and then we also want to know what kind of volume traffic is going to be sent to the Syslog servers.

You may have to deploy more than one syslog server, especially if the one that you are deploying is being overwhelmed and it's taken on the burden of multiple devices, especially if those are really busy devices.

For example, if you were to have a Cisco ASA Firewall and it were to send traffic to a single syslog server that might be fine, but if you had 5 ASAs and

they're all reporting to the same Syslog server, that doesn't have a lot of storage space or that has to continue to archive that off. It doesn't have a lot of RAM to process these events, well then you can run into some issues.

Let's take a look at some of the considerations or management guidelines. The first protocol to mention here is NTP, which is our Network Time Protocol. What do we want to do with it? Well, we want to keep our clocks on our devices synchronized.

Not only on network devices but also host devices, really everything that's touching the network, we want to have a common time source that we're all pulling from so that we all have that same time. We want to enforce our password policies and make sure that is consistent. Ensure that we don't have one group that doesn't have to update their passwords while another group does.

That's not a good password policy, in that case, so we want to make sure we're doing that. We also want to think about our out-of-band management. Perhaps if we have out-of-band management, we want to make sure that we're using the highest level of security possible on that out-of-band network, making sure that we don't run the risk of passing insecure management protocols like SNMPv1 or Telnet traffic over a production network where the traffic shouldn't be seen.

We want to use SSH if we're going to be sending traffic in-band. There's a difference between in-band and out-of-band. In-band means we're using our network resources to pass management traffic, while out-of-band means we have a separate network that's used for management.

We want to look at the two of them and ensure that if we are using the production network, if we're doing something in-band, let's encrypt that just so our traffic doesn't get captured by the wrong people and make it into the wrong hands.

We also want to use SSL if we're going to be in-band. Then we also could consider when it comes to these managed devices, should the connection or should that method of access always be enabled, something that's always on, or should this be something that we enable temporarily.

We allow the temporary connectivity and then make our changes, and then we disconnect, and it doesn't leave it wide open to the outside world. One

possible method here would be to use IPsec. We could use IPsec to terminate a VPN connection, and then only allow connectivity from the VPN address pool. This way, the connection is not always exposed to the outside world.

For example, the management connection like SSH or SSL, we could do that through an IPsec session that needs to be established, and that gives us an extra layer of security. That's something that we need to determine. Do we need that channel for management to be open at all times, or is this just something that should be temporary. These are all some general recommendations that revolve around keeping our devices secure but also allowing us the ability to manage these devices properly.

Chapter 44 CPU Threshold, Netflow & SNMP

We already been talking about critical components of our devices, so the memory and the CPU utilization definitely come into play here. One of the features that we can enable is called CPU thresholding. CPU thresholding uses a simple command process CPU threshold to enable a rising threshold or a falling threshold, and an interval in which if we either exceed or if we fall below these values, we would expect to receive an SNMP notification.

In that SNMP notification, it's going to have the top CPU users, so we can get an idea of what's making that CPU rise above the threshold for that interval. To enable CPU thresholding on Cisco devices are easy. First, you type the command “enable” then “config terminal” and we can run a command on the current CPU usage by typing “show processes CPU”.

We have to specify the type of thresholding, and the different types we have are, interrupt, which is the interrupt level utilization, we have the process, which gives us process-level utilization, or we can take everything and do consideration with the keyword total.

One of the things that we do need to make sure of is that we are aware of the interval values, and those are between number 5 through 86400. 86400 seconds is about 1 day. If all of a sudden, our utilization is in the dir, then the router, for example, is not doing anything at all, and that could indicate another problem. If we have less than 5% utilization over 5 minutes, something is not normal, and we want to be notified.

You do have to have SNMP traps configured for this to function, but this sets the thresholds, and that's what we want to just giving us that extra information to know if the device is acting the way it should or not. How does this fit into security? Well, in security we need to make sure that the infrastructure is protected.

If somebody were to have some kind of process or if somebody were to launch a Denial of Service attack or some kind of an ICMP attack, that attack against our router could cause the CPU to spike, in which case this threshold could let us know that something is going on. We need to make sure that the infrastructure is secure. Let's take a look at Netflow.

When it comes to Netflow, we're not concerned with how they get configured because these are things that you should already have some configuration

ability for. Instead, these are the things that should be considered. We want to focus on the recommendations that revolve around security.

Netflow is a protocol that's used to take flow information and then export that off to an external device, which we call a Netflow collector. Imagine a scenario, where we are going to enable Netflow on network switches, and we were going to be looking at flows or ingress traffic into those devices. What do we need to be concerned about? Well, let's start with the path that Netflow information would be exported.

Netflow export is a one-way flow, and it’s using UDP. We need to think about the path the Netflow information would take, and we need to think about what kind of information is going to be carried, because it’s something that needs to be secured.

In some cases, maybe it's not something that needs to be secured other than just letting that Netflow information travel. We would assume a certain level of security on the core devices, depending on how the network is built out, and so maybe we don't have to put information into a VPN tunnel.

But if the Netflow server is off-site, maybe located at the Datacentre, so if it’s across a Wide Area Network or WAN, and we're talking about one of our branches sending Netflow back across the WAN. And that WAN happens to be a public internet, well then in that case maybe we do want that traffic to be encrypted just because we don't want people to see our flow information and be able to learn our traffic patterns. We also have Management Access Lists or ACLs that we can further consider using and think about where we would want to put our Management ACLs within a network.

The last thing to talk about is SNMP. SNMP is based on a couple of different things. We have a network management server, we have a trap agent, and we have MIB. The SNMP server must be running specialized software to be able to receive SNMP information. Then, we also need to have an SNMP agent on the network devices such as on the Firewalls, or the edge routers to get a management information base which would be used to send information over to the SNMP server, or so that the SNMP server would be pulling that information.

These agents can send message that are unsolicited. The way that SNMP authenticates, depending on the version, and is called a community string for SNMPv1 and v2. We also have read-only community strings, and they can

get information, but they can't set information. Then we also have read-write community strings, and they can set information as well.

If you have set access, that's like giving you privilege mode on the router, that offers you all-access, and that may not be a good thing. SNMPv1 is a model based on what's called “noAuthNoPriv”. It's just community strings. There's no authentication, and there's no privilege control, it's just a plain-text string. That string, if it's clear text and somebody gets ahold of it, you're in trouble.

SNMP version 2c is also a “noAuthNoPriv” with a community string. It doesn't do authentication, it just does string matches, and it's not the most secured either. SNMPv3 has a couple of different options. It's got “noAuthNoPriv”, “authNoPriv”, and “authPriv”.

Using SNMPv3, we can use usernames, we can do “authNoPriv”, we can do MD5 and SHA, secure hash algorithms, and that provides an HMAC, which is a hashed message authentication code that's used for authentication. AuthPriv it's the most reliable level that you can get. It's authenticated with DES or 3DES or AES encryption, in addition to the authentication that uses the HMAC. Therefore, as a general recommendation, we want to make sure that we use SNMPv3 anytime we're managing devices.

Chapter 45 Control Plane Policing

The control plane of our devices is where routing protocols and device management happens, so this is part of the device's hardware and software that supports those features. It'll process packets that are sent to machines, and then it'll process packets that the router or the switch originates itself.

Anything that gets sent from your router comes out of the control plane, and anything that is destined to your router gets sent up to the control plane. We have three different planes that we generally work with. We have the control plane, which is what we're discussing here, and then there's the data plane, and then the management plane, but now we're focusing on what the control plane does.

It's crucial that we protect the control plane, because if someone can get in there and corrupt the routing tables, then it's easy to create a man-in-the-middle attack so they can redirect our traffic to go somewhere that it shouldn't, and then they can capture that traffic. Thus, we must protect this area of the device.

We want to make sure that we don't have too much load on the CPU, we want to make sure that we don't have different services that fail because somebody is hammering the control plane, so this is important. When somebody telnets into a router, we can apply an access list on it, and that would be an inbound access list or ACL, so traffic that comes into the router would have to pass through that inbound ACL.

The same thing is true for the interfaces. The interfaces can have an inbound ACL, and traffic has to pass through those inbound ACLs before they hit our router, and our router makes any decisions on how to move that traffic. Anything that's going to the control plane, any traffic coming in that are going to the control plane, it has to pass through any inbound ACL first.

That would mean either the inbound ACL on an interface or the ACL on the VTYs. When it comes to configuring Control Plane Policing, it's generally considered an advanced topic. Still, to give you an overview of the three main configuration elements, those are a class map, a policy map, and a service policy.

We want to protect the control plane, and we generally will police traffic so that we can control the rate that traffic is coming into the control plane so that

we don't end up spiking the CPU because we get hammered all of a sudden. The first thing that we would need to do is create a class map, and that class map is going to identify what traffic we should be looking at. Typically, a class map is going to reference an access list. An access list is a filter, so it's a way to match on packets, and the class-map will reference the access list.

Chapter 46 Authenticating Routing Protocols

When it comes to routing protocols, they communicate crucial information with one another, they exchange information that has to do with network connectivity, as well as how we reach different destinations in the network, and they build forwarding paths between destinations. If we don't have routing protocols and we have to configure static routes, then this becomes problematic as our network scales, so routing protocols are an essential thing for us.

When we want to prevent an attacker from being able to participate in our routing domain, there's something that we do. We authenticate the updates. Mostly, we use a cryptographic authentication, which creates a hash of our update, and then we send that hash along with the update. This does require a shared secret to be configured on both ends.

You have to ensure that the shared secret is a secret, because if somebody in the path were to learn it, then they could generate their updates, and we would trust them. That would be an issue. These hashes that are generated by the shared secret is called a Message Authentication Code, or a MAC, and sometimes you'll hear these in IP sec referred to as an HMAC, a Hash Message Authentication Code.

Make sure that those are kept secret and configured manually on both sides, and keep in mind that this is not encrypting the update, it's merely authenticating the update, so there's no secrecy or privacy or confidentiality here. If an attacker is in the path and they grab that update, they have the hash.

These hashes, if they're strong then we're good, but if they're not strong then they could be susceptible to a brute force attack, and if that's the case then we run into some problems. If somebody were to launch a brute force attack, then it's just a matter of time before they guess the password or they are able to crack that password, then we are in a whole lot of trouble.

We generally don't route down at the access layer, unless we're doing layer three out to the access layer. At the Core, the Distribution, the Data Center, we should enable some routing protocols, and if you work with OSPF known as Open Shortest Path First, you have to configure OSPF by defining a

keychain. When you specify this keychain, you have set its cryptographic algorithm to use HMAC SHA256.

SHA stands for Secure Hash Algorithm, 256 is the bits, 256-bit Secure Hash Algorithm. That is considered a secure way of doing it. It is new in terms of routing protocol authentication. Older routers don't support this, in which case you'll probably have to go back to an MD5 authentication or Message Digest 5, in which case you would just set the cryptographic algorithm to MD5 and be done with it.

It's not as secure, but it's not supported everywhere with SHA256. Once these keys are configured, one thing that you want to know is that these keys, by default, they're always valid, and they never expire. If you're going to get a little more granular, you can set their validity times and have them expire, and you can roll them over or rotate them. This would give you more security. Once you define this keychain, which is the first thing that you would do with OSPF, you have to apply it to the interface directly, and then you have to verify it.

With EIGRP authentication, the configuration is very similar. We can use the SHA256, and if we use that, we have to use a unique configuration where it's named in our authentication configuration. Next, configure a keychain, reference that key inside the keychain, and then apply it to an interface, so it's very similar in terms of configuration.

Chapter 47 802.1X Authentication

802.1X is heavily used in our campus environments and wireless networks. Why do we need 802.1X authentication process? Well, imagine the following scenario. Imagine that we have a couple of clients, and one them is a Windows 10 based OS-s, and then we have an iPad. The iPad is going to want to connect to the network, and the Windows 10 machine is also wants to connect to the network, but we want to protect our network from unauthorized devices gaining access.

If a user were to bring in a computer and plug into one of the publically available ports, we want to have some control over whether or not we're going to allow them to get to certain areas of the network. For example, are they going to be able to send traffic from an unauthorized or an unknown Windows machine that connects into our network?

Are we going to let them send traffic across the network and start connecting to our servers? Is that something that we want to have happening? In most cases, we want to have a lot of control, and this is why we use 802.1X.

802.1X is an IEEE developed protocol, and the protocol was designed to do port-based authentication. When we say port-based authentication, we're talking about authentication on the interface where a user would connect. If you remember, on Cisco network switches the switch ports are enabled by default.

Occasionally, engineers don't follow recommended configurations, and they leave these ports enabled. But, if we use 802.1X technology, it'll give us an extra layer of security where we can leave that port enabled, but if somebody does connect to it and the link becomes active, well now we can find out who they are, and decide what they're allowed to do.

It's going to involve the users that are connecting via our wireless network that could not be an iPad, but it could be a Windows 10 machine that's not wired, it's just coming into the network. It could also be a Mac, trying to connect into the wireless, or it could be a corporate wireless device that is trying to connect to the network. Maybe they know the password to the wireless network.

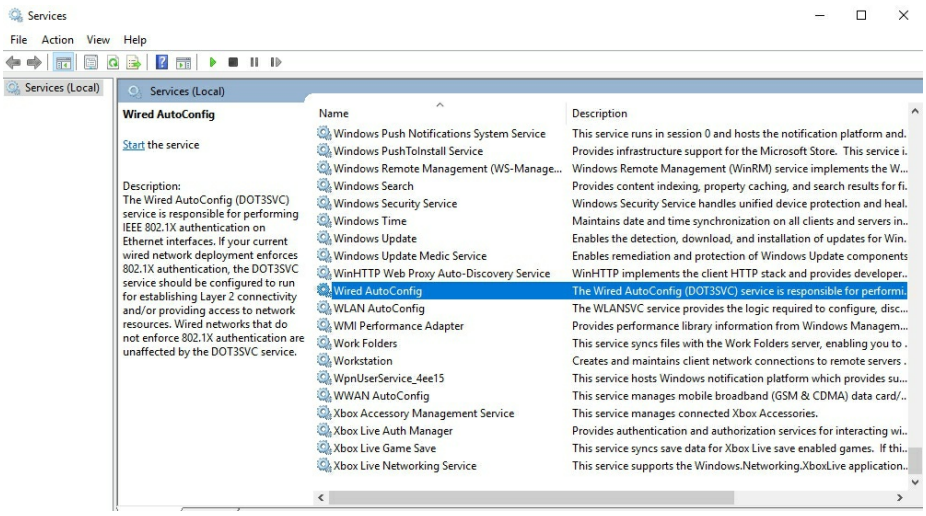
They perhaps know the WPA2 key because they're a user within the organization. Just having that key on the network is probably not going to be

enough so that we can use enterprise-class authentication instead, and that's where 802.1X would come in. One other thing to point out is that we're going to have a couple of different elements.

There are few terms that you are going to see in an 802.1X environment, so let’s discuss those first. The first term is called “Supplicant”. What is a supplicant, and why is it important to us? Well, a supplicant is a software that is installed on a client device. It could be a Windows 10 machine, it could be an iPad, but either way, the supplicant is the client-side software that supports 802.1X communication.

This could be what we call a native client. The native client would be just working with Windows 10 and enabling authentication inside the Windows 10 network adapter. On a Windows machine, if you want to configure the supplicant, go into the network adapter, go to Properties, and find the Authentication tab.

If you don't see the Authentication tab, it's because you need to enable it in your services. So type services, and go all the way down to the bottom, and look for Wired AutoConfig.


Services
File Action View Help
Services (Local)
Services (Local)
Wired AutoConfig
Start the service
Description:
The Wired AutoConfig (DOT3SVC) service is responsible for performing IEEE 802.1X authentication on Ethernet interfaces. If your current wired network deployment enforces 802.1X authentication, the DOT3SVC service should be configured to run for establishing Layer 2 connectivity and/or providing access to network resources. Wired networks that do not enforce 802.1X authentication are unaffected by the DOT3SVC service.
Name
Description
Windows Push Notifications System Service
This service runs in session 0 and hosts the notification platform and.
Windows PushToInstall Service
Provides infrastructure support for the Microsoft Store. This service i.
Windows Remote Management (WS-Manage...
Windows Remote Management (WinRM) service implements the W...
Windows Search
Provides content indexing, property caching, and search results for fi.
Windows Security Service
Windows Security Service handles unified device protection and heal.
Windows Time
Maintains date and time synchronization on all clients and servers in..
Windows Update
Enables the detection, download, and installation of updates for Win.
Windows Update Medic Service
Enables remediation and protection of Windows Update components
WinHTTP Web Proxy Auto-Discovery Service
WinHTTP implements the client HTTP stack and provides developer..
Wired AutoConfig
The Wired AutoConfig (DOT3SVC) service is responsible for performi.
WLAN AutoConfig
The WLANSVC service provides the logic required to configure, disc...
WMI Performance Adapter
Provides performance library information from Windows Managem...
Work Folders
This service syncs files with the Work Folders server, enabling you to .
Workstation
Creates and maintains client network connections to remote servers .
WpnUserService_4ee15
This service hosts Windows notification platform which provides su...
WWAN AutoConfig
This service manages mobile broadband (GSM & CDMA) data card/..
Xbox Accessory Management Service
This service manages connected Xbox Accessories.
Xbox Live Auth Manager
Provides authentication and authorization services for interacting wi..
Xbox Live Game Save
This service syncs save data for Xbox Live save enabled games. If thi..
Xbox Live Networking Service
This service supports the Windows.Networking.XboxLive application..

This Wired AutoConfig service is responsible for performing the 802.1X authentication on your Ethernet interfaces. If you open this up, on Windows by default, you can see that this service is stopped, and it's set to Manual. If you're going to be using the native supplicant that's built into Windows, you're going to want to change this to Automatic and then start it. Then once you do that, you can go ahead and close the Services tab.

Now, if you go into the Properties for your Ethernet adapter, now you will see that you have the Authentication tab, and now 802.1X authentication has been turned on. This is the native supplicant that is built right into Windows. We also have native supplicant capabilities on our iPads, on our iPhones, and our Mac computers. On Mac, or Linux machines, they also have supplicant capabilities, and this is also built right into the operating system.

Knowing that we have a supplicant in our devices that is native, we also can install a supplicant. That supplicant that we install, we would install on a Windows machine, or we could also install it on a Mac, and that would be the Cisco AnyConnect Client. The AnyConnect client will give me VPN capabilities. That's what most people are familiar with AnyConnect for, but it also has the Network Access Manager, which provides me with 802.1X capabilities. That's the first term that we want to understand.

The next term that we want to understand is an Authenticator. What is an authenticator? Well, it depends on the scenario that we're working with. The first authenticator typically is a network access switch. This access switch could be the authenticator for the Windows 10 client who's going to be attempting to connect to the network.

Another authenticator could be a wireless access point, that would be authenticating an iPad that connects. Generally, an access point is going to be talking back to the wireless LAN controller, and that wireless LAN controller is where the configuration of the authenticator would talk place.

Authenticator is going to be a device that supports 802.1X. Not all access switches are supporting 802.1X, but there is always a compatibility matrix that you can look at specifically on the device details that you would want to use.

So an authenticator would either be a network switch, a wireless controller, but it could also be a firewall, such as a Cisco ASA that could act as an authenticator, and a Cisco router can also act as an authenticator. Then the third term that we want to understand is the Authentication Server. The Authentication Server is none other than Cisco's Identity Services Engine or Cisco ISE. The Identity Services Engine is the authentication server that's going to store all of the information about what users are allowed to do, so the policy will all be defined right there.

That doesn't mean that ISE has all of the user's credentials. In fact, in most

cases, there is an additional server that is running AD or Active Directory. If that's the case, then if a Windows 10 user tries to authenticate with ISE, ISE may go to an external database and asks AD about the username that’s logged on to the Windows 10 machine.

First of all, is that a machine that we know, is it a domain machine, and then is the user account and their credentials valid? And then AD can come back with an answer on that. Those are the three terms that you need to understand in 802.1X authentication. We have an authenticator, we have a supplicant, and we have an authentication server. The next thing that we want to look at is the protocol communication that's happening in this authentication process.

We have supplicants that are talking to authenticators, and we have authenticators that are talking to authentication servers, we have authentication servers that are talking, perhaps, to Active Directory, so all of this communication is happening through various communication protocols. The first protocol that we want to discuss is EAP.

EAP stands for Extensible Authentication Protocol, and this is a framework that we use with our wireless networks, or with our Ethernet networks to provide and perform that authentication. EAP provides a transport, and it also gives some keying material that can be generated by various EAP methods.

When we say EAP methods, what we're talking about are different ways that this framework can be used. Some EAP methods using simple passwords, and some EAP methods using digital certificates. We have these EAP methods that are defined in IETF RFCs, and they would include something like an EAP.

EAP is the protocol framework, but we also have EAP-MD5. MD5 stands for Message digest 5, and EAP-MD5 is something that would use a password, so it's minimal security. It does an MD5 hash, it is vulnerable to dictionary attacks, so it's not necessarily the best method to use, but it is a method that we can use. Another technique that we have is something called EAP-TLS, Transport Layer Security. Transport Layer Security, or EAP-TLS, we're going to find this defined in RFC 5216. This is an open standard. It uses TLS.

TLS is what has become the successor to SSL. When you think about that, you're probably thinking about your connection to the bank, that is secure, it uses Transport Layer Security or digital certificates to do authentication of your credentials. That's exactly what's happening 802.1X.

In the client-server implementation with EAP-TLS, it's supported through several vendors. Cisco supports this for sure, but other vendors also do. Apple supports EAP-TLS, Microsoft supports EAP-TLS, which means that we can use it with their clients there as well. It is natively supported in Mac OS X 10.3, and above, Windows 2000 and above supports it, XP and above, so it's in our standard operating systems today. That's just another method we have.

We also have PEAP or Protected EAP, and that acronym PEAP is another method that we can use. The way that PEAP works is that we have the protocol encapsulates EAP, that Extensible Authentication Protocol, that framework in an encrypted and authenticated TLS tunnel. So we're tunneling our EAP sessions, and that's how we are protecting those EAP sessions.

PEAP was something that was developed by Cisco, and Microsoft supports it. Windows machines have the option for Protected EAP because this is something that is built right into the native client, but we want to understand that this is just another method that we can use to communicate. That's the EAP authentication.

So, when it comes to EAP authentication communication, it's going to take place between the supplicant, such as the Windows 10 that has PEAP capabilities, and the communication is going to be right there. This is where our EAP is going to happen. What about if we're using something like a laptop or an iPad or something like that, and we're going to be using 802.1X?

Well, now the EAP session is going to be sent to the access point, and then put into that CAPWAP tunnel that heads back up to the wireless controller, which would be the authenticator in this case, and then we would switch to another protocol at that point, but that's where we're going to see that. It's not going to be between the iPad or the Windows client and the access switch where the AP is connected, because wireless traffic gets put into a CAPWAP tunnel in the lightweight access point configuration and it's sent over to the wireless controller. That's one part of the communication.

The other part of the communication is what we call RADIUS. RADIUS is an IETF standard method of communicating with an authentication server. We already discussed the AAA authentication. We can use TACACS+ or we can use RADIUS. With 802.1X, we use RADIUS, so where is the RADIUS communication going to take place?

Well, it's going to be from the access switch over to the authentication server. Remember that Cisco ISE is an authentication server. Back to RADIUS, it has a couple of message types that we use. We have Access-Request messages, so you could think that the access switch is going to send an Access-Request over to ISE and ask about a user that wants to authenticate, so the switch is asking ISE if it can allow the users on the network.

We also have an Access-Reject message. So ISE might come back and say, “no”, that user is not authenticated, here's an Access-Reject, in which case the switch port where the user is connected would be disabled. The user would not be allowed to connect, and perhaps an EAP message would be sent to the client and say: “you're not a valid client, you're not allowed to connect”.

The other type of message that we have is called an Access-Accept message, and that RADIUS Access-Accept message could be sent from ISE back to the access switch and say, “yes, this user is authenticated”. Alternatively, we could have ISE talking to the wireless controller with Access-Reject and Access-Accept messages, assuming that the access point is where the communication is coming in through.

Maybe the Windows machine is using a wireless connection into the AP, that's being put in CAPWAP back to the controller, and then the controller is talking back to ISE. That's an overview of the RADIUS communication, and these are the crucial protocols that we need when we have 802.1X. If we're using something like an EAP method that is using certificates or EAP-TLS, we might also be using ISE to talk back to Active Directory. Then if that's the case we can do our authentication back at ISE, and there is some protocol communication that would have to happen there. For this purpose, we would integrate ISE with Active Directory, which would mean that ISE would join the AD domain, and it would communicate on the standard Active Directory ports from ISE to the Active Directory server.

Now that we know how the communication works, we also know the different protocols that we can use, once we start authenticating users, we have to think about what's this going to be allowed to them and what’s not.

But what kind of actions are possible when we're performing 802.1X authentication? Well, there's several actions that we can take, but we're only going to focus on a couple of them that'll be important. The first is what's commonly used called a DACL, or a Downloadable ACL.

A Downloadable ACL is something that can be configured in ISE, and the format of the ACL looks kind of like a standard access-list so that it might say something like Access-list 10 permit IP traffic from any source to a particular server address. When a Windows machine authenticates and ISE notices that there is a downloadable ACL applied to that user's policy, that ACL is downloaded to the access switch and applied on the switch port where that Windows 10 machine is connected.

That's only one action that we can take for applying policy when we're doing 802.1X authentication. The other method that we'll talk about is VLAN Assignment. If a user want to authenticate and the policy in ISE says that once a particular user authenticates, associate them to VLAN 10. Maybe they started on VLAN 1, but now we're going to put them into VLAN 10.

This is going to mean that the user is going to have to have the port bounced so that they can do another DHCP request and get an IP address that's on VLAN 10, and within that VLAN 10 address space, but we can use this to freely move people from VLAN to VLAN if we need to, just based on how they're authenticated.

The other thing that we could do, which will be more important to us when we get into BYOD, which stands for Bring Your Device, is that we can do apply a Redirect URL. Redirect URL is what we typically see used in guest access. For example, if a user tries to authenticate, the authentication fails, and ISE sends a Redirect URL back to the switch. Then, the switch has to have the HTTP server running, and the command for that is “ip HTTP server”, and if that server is running, then the access switch can send a Redirect to the Windows machine.

That will then cause their browser to launch and redirect them to a guest portal, and then from there, we can go ahead and authenticate that user using a guest account that they've been provisioned with. This is something that's a little more commonly seen in the BYOD scenario, but we can use that for fallback access on switches, send them back into a guest portal and allowing guest access as well.

We can also apply this very commonly to our access point configuration, where mobile devices come onto the network. For example, if you go to visit a store, you connect to the local Wi-Fi, and a window opens up and it wants the guest account credentials. You type in the guest account credentials, and

then that window disappears, and then it shows that you're connected to the Wi-Fi.

These are some of the options that we have, but there are many more in ISE. These are the actions that we can take, so now, let's bring this all together and talk about the process flow from start to finish when a user comes onto the network and wants to have access.

There are many different ways that we can use to implement the 802.1X authentication process flow, and we're going to take this from the perspective of this Windows 10 machine. The first thing that happens is that the user will plug the device into the network. We connect, and as soon as that link goes active, the end device is either going to send an EAP-Start message known as “EAP-Start”, or the switch is going to send an EAP-Start message.

At this point, they will start the process of EAP authentication using whatever the selected EAP method is. That could be EAP-MD5, it could be EAP-TLS, it could be EAP-FAST, it could be PEAP, so there's a whole number of them that we could choose to use here. As we collect information between the Windows 10 machine and the access switch using EAP, from the access switch, we are going to initiate an access-request over to ISE.

This is a RADIUS Access-Request. The ISE is going to do a couple of things. ISE is going to look at an authentication policy, and he's then going to see if the user authenticates. If the user authenticates, he's going to roll into the Authorization Policy. For the Authorization Policy, we're going to be looking for a match. That would be based on several different criteria.

It could be what kind of operating system are you, what access switch did you connect to, what group are you in when you authenticated, but let's just say we land on an authorization policy for employee access, and it says that a user who is in the Marketing Group is going to be allowed access with a DACL or a Downloadable ACL, and that DACL is “permit ip any any”.

In that case, ISE is then going to send an Access-Accept message back to the network switch, and this is an Access-Accept message, and it's going to say a DACL = PERMIT-ALL. The access switch is then going to see that, and it's going to send an EAP success message to the Windows user, which essentially lets the native client know that the authentication has been successful, and the switch port is going to become active.

In addition, on that switch port, we're going to apply the PERMIT-ALL access list, which will then control that Windows user's access out to the rest of the network. From this point, the user has full access to the network, and after a period of time we'll do a re-authentication just to make sure that nothing has changed. There is a lot more that we can do, but this is general 802.1X authentication flow.

These concepts are important because they're also used in the BYOD deployments that we see so much today, and in our next chapter we're going to look at the BYOD process and the components that are involved in that type of a scenario.

Chapter 48 BYOD Security

BYOD stands for Bring Your Device, and this is a prevalent topic these days, so we're going to start by discussing some of the challenges that we see today, and then we'll do a bit of an introduction to BYOD. We're going to take a look at the Single SSID BYOD deployment, and then we'll take a look at the Dual SSID BYOD deployment, and we're also going to talk about MDM, which stands for Mobile Device Management.

Let's start with the challenges. Nowadays, things are different than they were 5 or 10 years ago. There's this new paradigm, and the way that we work has shifted to an end-user device is the primary device that most people want to use. It's often been said that user devices that we can buy at the store now have surpassed the capabilities of what IT departments have deployed in an organization.

Whereas we would maybe get a laptop issued to us in the past, and the IT organization strictly controlled that laptop, people want to use their tablets now, people want to use their smartphones, they want to use applications that they can sign up for on the internet, such as software-as-a-service applications, and they want to use these to get things done. This is a new shift in networking as well as in security that we've experienced over the past several years. Still, it's very different than how IT organizations were built previously.

Therefore this becomes a challenge for an IT organization to get this type of network deployed and secured to ensure that people can bring their own devices. Still, even they're bringing their own devices, we should have some control over them. That leads us to the next challenge. Not only does an IT organization have to be able to adjust to this shift in how networking is done and how employees are handling work, but we have to worry now about how we onboard these devices.

What is onboarding? Well, fundamentally, what we mean here is how we take a device that does not belong to our IT organization and add it as a trusted device on our network. The question is, can we do that, can we trust it? Well, yes, with our modern methods of control such as 802.1X as discussed previously, the ability to find out who a user is, we can decide what they're allowed to do using Cisco ISE as the keystone product for this whole

device onboarding method, we can do this successfully, but we have to be concerned about it and how it's going to be done.

Device onboarding is an issue that we need to deal with. Another reason that this becomes an issue is that the way that we implement protocols for EAP may not be supported within the client that somebody is trying to bring into the organization, so there's just another challenge that we face when we onboard a device.

Lastly, our third challenge has to do with controlling that device. IT organizations don’t own these devices, but we need to have some kind of control over it if it's going to be accessing resources. Let's go back in time and remember when we would set up a VPN connection from home, and that became more and more popular.

People wanted to work from home, we would set up a VPN connection, we'd allow them to connect in from home, and then once they were connected in, we would use an access-list to control what they could do as far as what kind of traffic they could send, but let's just analyze this for a moment. You have an endpoint that is accessing the network, and you're controlling its IP address as far as what it can access, but what about the data that it is accessing, where is that data stored, and is that data able to be shared? See that's a problem.

That's been a problem for a long time. It's even more so an issue when you have a mobile device such as an iPhone. What if the iPhone is lost, and there is personal data sitting on that device, how do we control it, how do we lock it down? Well this is where our MDM solutions come into play to assist with this challenge that we have.

We have a number of MDM solutions that are third-party built that integrate with various environments. We're not going to go into the details of those, because a Google search will get you an idea of who those are, but we're more so concerned with just the fact that this is something that we need to be able to deal with. These are the challenges that we see in this new environment that we work with where people want to bring their own device. Let's go into a little bit more detail on the device onboarding. People say terms such as onboarding or device onboarding, but what exactly is going to be involved in device onboarding? Well, we're going to have to deal with credentials.

The first question is, where are these users credentials going to be checked, where do they exist, and how do we make all of that happen? Credentials in an organization generally live in a database, such as in Active Directory. Active Directory is Microsoft's directory database. We could also have an LDAP database that we might be using, but Active Directory is probably the most common.

As we onboard a device, we need to have a way that we can take credentials from an end user's device, either providing us with credentials by typing them in, or by having a certificate that’s on that device. We need to be able to take that information and pass it over to our authentication server, and that's a term that we learned in 802.1X. We know that the authentication server is likely going to be a Cisco ISE or Cisco's Identity Services Engine. We need to be able to have that communication between those devices so their credentials can be validated.

The next thing that we need to think about with device onboarding is how we provision the supplicant. Remember, a supplicant is a client that supports 802.1X authentication, and 802. 1X authentication is used in our BYOD environment so that we can take those credentials from the user and then pass that information over to ISE.

Well, when we provision a supplicant, this might involve provisioning them with additional information, additional certificates, we might even provisioning certificates so that they can authenticate. We have to provision the supplicant so that it knows exactly what kind of communication it should be using, such as EAP-TLS. Still, we also need to push down certificates to the device so it can be provisioned with those certificates to perform authentication.

Another aspect of device onboarding is the end-user registration. In a Cisco environment, for example, this is easy as far as end-user registration. It's something that they can do on their own. As users go through the device portal, we can take their MAC address, we can register that end user's device, and they even have a “My Devices Portal” that they can go back into later on.

For example if users were to lose their phone, they can go into their “My Devices Portal” and they could edit their device as a lost device, and ISE it's going to ensure that device is not allowed on the network anymore. It blacklists it. This is device onboarding, and it's one of the biggest challenges

initially to deploying a BYOD solution, but let’s get a little deeper into this.

As we deploy BYOD, there are two different architectures that we will generally see, and one being more popular than the other. Those are considered the Single SSID or the DUAL SSID architecture. SSID is the Service Set Identifier, and that's what we use to advertise a wireless network. When you go into an office and look at the available wireless network, those names are known as Service Set Identifiers or SSID-s.

Back to BYOD deployment, there are two ways of doing it. We can do a single SSID, and on that single SSID we have corporate users, and we have BYOD users, or we could split it into two or three SSIDs. With a single SSID, you're provisioning of the supplicant, as well as network access, is all happening on the single SSID. In contrast, with a dual SSID, we're going to separate the provisioning and the network access.

You'll connect to one SSID, and you'll get provisioned, and then after you're provisioned your client, the supplicant knows that it needs to disconnect from the first SSID and reconnect to the provisioned SSID to get BYOD access. With a single SSID, we just have a secure one SSID implemented, where maybe we're doing WPA2, and we have that passphrase that we have to enter to connect to it.

That can be problematic for people who want to provision as they need to enter that information to gain network access. In contrast, with a dual SSID, it's similar to that traditional approach that we've discussed. We have a protected SSID for our employees, and then there's an open SSID for guests.

This is very common. Most organizations that are large enterprise networks that you go into, you're not going to see your guests on the same wireless network as the employees, they'll be separated off, and that's a pervasive approach. We would use an open SSID, the one that we generally use for guest access, and we can use that SSID to provision our employees who have brought their own devices in. Then once they have provisioned those devices, we can go ahead and send them over to the protected SSID with their equipment that's been registered in our BYOD environment. That's what happens there, and we use that open SSID. With a single SSID, it doesn't work for guests because the guests can't connect and put in guest credentials.

They need that secure passphrase to connect to our WPA2 environment, and that just doesn't work. However, the dual SSID environment is good for

guests, and it is suitable for employees, and this is an excellent method of separating the access and still being able to register an employee who has brought their own devices. The single SSID solution is not a common solution, in fact, it's not recommended. The dual SSID solution is what's recommended, and that's the more universal solution that we see in our enterprise networks today.

Let's walk through the process that takes place with an iPad that wants to connect to our access point and then access the network, using a BYOD scenario. When it comes to dual SSID authentication, we have an access point, and the access point is advertising more than one SSID. We've got Guest SSID and Employee SSID, and let's imagine that there is an iPad user who is going to associate to the access point on the open SSID, which is Guest.

Once they've done that, they open up a browser, and they're going to get redirected to ISE. ISE is going to deliver the web page back to the iPad. If we were to look at the IP address or the hostname of that URL that they've been redirected to, it's ISE, and this puts them into what we call CWA or Central Web Authentication.

As they go through Central Web Authentication, they provide their user credentials that are part of Active Directory, because this is an employee that's brought their device into the office, and they want to use it. The authenticator authenticates them back to the employee's Active Directory, and then it's going to move them into a device registration portal.

This device registration portal is something that is being delivered by ISE, so they're going to be forwarded another registration portal. That's device registration, so they'll go ahead and register their device. This captures their MAC address and it prepopulates it into the registration page, and then they add their description, which users have to type in on their own, they submit it, and at this point the supplicant gets provisioned and a certificate gets installed, and that's all going to happen on the iPad.

They'll be provisioned, and once that profile is provisioned and that certificate is installed, the user disconnects off of that open SSID, and they go ahead and connect to the employee SSID using the information that was provided in the supplicant provisioning. That's the high level of what's going on when a user authenticates in a BYOD situation back to ISE.

The last thing that we need to discuss here is an MDM solution. MDM stands for Mobile Device Management, and this would be a cloud-based service. The MDM server could be AirWatch or Mobileiron or something similar. When using an MDM, ISE ends up being registered with that MDM solution, and whenever a user tries to connect, we can have rules in ISE that are going to reach out to that MDM server and ask for the credentials, or if the information that this user has provided are matching up to MDM's policy.

From there, the MDM server can let us know whether or not that user is allowed to connect, whether they have the right configuration on their phone. This MDM solutions are subscription-based servers, and the MDM server is going to communicate back to your devices. It'll have them install software on end devices that can track and register into the MDM server.

Using an MDM server is going to give you more granularity as far as being able to control exactly what's happening on the device that an end-user has brought into the environment. MDM is an important concept to understand because we need that control just like we would have control over a corporate asset, and be able to see that's installed and what specific applications we do not want to allow.

For example, maybe we don't want to have an FTP client on that iPad that could potentially FTP off the data somewhere else, or we could also control or even block gambling websites or torrent websites. But this is just an example of how we might use an MDM server in conjunction with ISE to provide a more gritty BYOD environment.

Chapter 49 Introduction to Firewalls

Let’s begin by briefly discussing security on devices and what a firewall might do, but not a lot of detail there. Firewalls are designed, to prevent the spread of a fire if the house were to catch on fire. Firewalls in a network are doing the same thing, they are creating zones, and those zones give us isolation, and we can control traffic between these zones. The most polular and trusted firewall indeed is Cisco ASA, might hear referred to as the ASA, or Adaptive Security Appliance.

More recently you might hear “ASA with FirePOWER Services”, or some people might refer to it as the Next-Gen Firewall, or NGFW. That's another common term for it as well, which is a virtual appliance, as well as physical appliances for ASA. These devices are firewalls, but a router could also act as a firewall. Keep that in mind, but I am focusing on the ASA here, but it doesn't necessarily have to be an ASA. So if an ASA or a router were acting as a firewall, it would be using something like Context-Based Access Control, or what we call CBAC.

What we might be doing with these devices is something called a Zone-Based Policy Firewalling, or ZBPFW, or just a ZBFW, Zone-Based Firewall, that's generally how it’s known. That's the overview of what a firewall does, creates zones, and gives us isolation, so let’s discuss how firewalls are implemented. Firewalls can be implemented on routers and switches, in which case the most basic level of capability is that they're filtering packets, and these would be packet-filtering firewalls.

For example using access control lists or ACL-s, is a simple way of creating a non-stateful firewall. That term “non-stateful”, we'll get into that in a little bit, but for now just as an example that ACL-s could be one implementation, where just a router or a switch with an access control list filtering packets. But, there are other implementations that we might see in a dedicated firewall appliance.

The dedicated firewall appliance would be something like a Cisco ASA. If you want to see the different devices that are available to us, you should visit:

https://www.cisco.com/c/en_uk/products/security/adaptive-security-appliance-asa-software/index.html

Here, you can find some small office, home office-type firewalls, and that would be something like the ASA 5500 series, and those particular devices

there are running software on top of Cisco hardware. The software is the ASA code, and those also come with what's called FirePOWER Services, so they have a solid-state drive in them that runs a separate operating system, a separate module called the FirePower Services Module.

Cisco Adaptive Security Appliance (ASA)

That's just a high-level overview on that hardware called the ASA, the Adaptive Security Appliance, which is a dedicated firewall appliance. In addition to that, other firewalls could be implemented as a very complex system where multiple devices and appliances are all combined, and they're all working with one another to provide firewall services. Very complex type of a setup, but the FirePOWER Services modules, which are handled by a FirePOWER Management Center, or FMC, that would be an example of a more complex integrated system that's providing firewall services.

It doesn't matter which of those implementations you chose as your firewall. There are a few requirements that they have to meet, however. First off all, they have to be resistant to attacks. They have to have an operating system that is not riddled with vulnerabilities themselves. They also have to be the only transit point between network zones.

To explain this, imagine the following scenario. Imagine that we have our headquarters, and in this topology, we have a firewall, and that firewall is a transit point between two different zones. Imagine that we have the internet on the left-hand side, and we have our headquarters network on the right-hand side, and our Firewall is seating right in between them and there's only two interfaces that separate them. The interface that is connected to the internet, we might call that the outside zone, or outside interface, and then the rest of the network on the right-hand side, we would call inside zone. So we have an inside and an outside zone. It's very clear when we look at which

zone would be trusted. The inside network, the headquarters side is more trusted than the internet side or the outside zone.

Because of that, we are able to control things a little better, we're able to make sure that we have proper security applied, and this is the requirement of a firewall, that it is the only transit point between network zones, in our case, the transit point between the inside and the outside of our organization. Also the configuration of that particular firewall, we're going to have to build that based on the organization's security policy.

Assuming that we are working for a customer, we're a network integrator, we want to find out what their security policy is, for example, what kind of internet traffic is allowed into their organization. For example, if they were to have a web server, and that web server is running in their datacenter, you're going to put it on a DMZ, if the web server were something that people needed access to, then we're going to have to make sure that the firewall is allowing traffic to come through from the internet, and then that it's going to be allowed to go through our network and get to that particular web server.

Usually, we don't do this, that's not good practice to put our public-facing type devices in the datacenter. What we would generally want to do is hanging right off of the firewall. We would configure something called a DMZ, just another interface that has public-facing web services, so we could put a webserver sitting there. Then we could allow access inbound to that web server without having to enable any connectivity to from the outside to the inside.

That would be a much better way to do things, and then that way, we're controlling the traffic that's in and out of our network. But that's another requirement, and we need to know what the security policy is for the organization so that we can build our configuration to match.

Then finally, we need to understand if the components that we're implementing, like our firewall, is part of a more substantial architecture or a more extensive solution. In this case, are we dealing with just a standalone firewall, or are we going to be controlling the FirePOWER Services module that's in that device that's part of a broader architecture, where we have multiple services modules throughout the network all controlled by a management center. These are some of the requirements that we need to find out ahead of time so that by the time we implement these technologies, we're

prepared, and we can wrap our head around what the customer is looking for.

Now, let's discuss each of the different types of firewall implementations. We have a couple of different implementations that we'll need to discuss, so I want to make sure that you fully understand packet filtering firewalls versus stateful firewalls versus proxy servers. Then we'll talk about Next-Generation Firewalls such as the ASA and what they do for us. Let's start with packet filtering firewalls.

Chapter 50 Stateless Firewalls

Packet filtering firewalls are what we would call stateless. What does that mean? Well, each packet that passes through the firewall is going to be filtered independent of one another, and we're not going to remember the state of any connectivity, we won't see how the connections might relate to one another, so there's no tracking of any of that information.

Each packet is individually controlled when we deal with a stateless packet filtering firewall. When we have a stateless packet filtering firewall, we're going to either permit or deny traffic based on just the information that's in the header. Every time we package up data to send on the network, we encapsulate it at each layer of the OSI model, by adding headers, and that header data has different information there that we can look at, so we can look at our layer 3 header that's going to have our IP address.

It'll also have protocol information in there and things such as time to live information or TTL. We can look at our layer four header, which includes port information, which we might relate that to something like web traffic, that would be TCP traffic on port 80. If we're looking at that type of information there, we might look at the TCP flags, so for example, if it’s an “ACK” or a “reset”. We can see all that with a simple access list. That's how these are implemented in the form of an access control list.

We could be looking for things like the SYN bit or the FIN bit to be toggled, but these are values that are inside the layer three and layer four header that we're able to match in an access control list, and those access control lists are going to handle packets one at a time, independent of one another. They're not going to maintain any state information or relationship between the different packets as to being part of a connection. They don't do that, because they are stateless firewalls. One great thing about these types of packet filters is that they are swift.

Chapter 51 Stateful Firewalls

Stateful firewalls are going to track a lot more information about the packets that are passing through them. By doing this, they're able to control the sessions. That information is going to be stored in what's called a state table, and for example, a Cisco ASA maintains one of these. Another example might be a Cisco router that is also running Zone-Based Firewalls or CBAC, and they're going to keep a state table as well. Both the ASA and IOS devices have these tables.

As far as the traffic passing through these devices, they are going to be based on the information that's in the state table. For example, when a packet leaves the network, generally, that's allowed, and we're going to allow someone to connect out to the internet. As the traffic leaves, we're going to add that information to the state table. Return traffic that's part of that session, and it should be allowed back in without being dropped, and usually, that's what'll happen.

We can have additional rules that are going to look deeper into the packet, looking at the application to make sure there's nothing malicious in there, but usually, if traffic is allowed to leave, and then we want to receive return traffic, so the state table would let that to happen. We can still implement access control lists because that's not a problem.

The access control list is going to decide as traffic comes into the interface, if is it allowed or not. If it's allowed, if the traffic gets permitted, then an entry would be created into the state table. If the traffic is not allowed, then it'll get denied, and it'll never make it to the ASA's stateful table or the firewall stateful table, and then we don't have to worry about tracking traffic that's not allowed. Access control lists are going to give us some additional control over unsolicited traffic that comes in perhaps from the outside, because that's the most common place that we see this. Thus using ACL-s, we're going to have to adjust certain protocols that a network channels during the stream. This is only an overview of a stateful firewalls, so let’s move on and look at Proxy servers.

Chapter 52 Proxy Servers

Another type of firewall technology that we see in use today is what's known

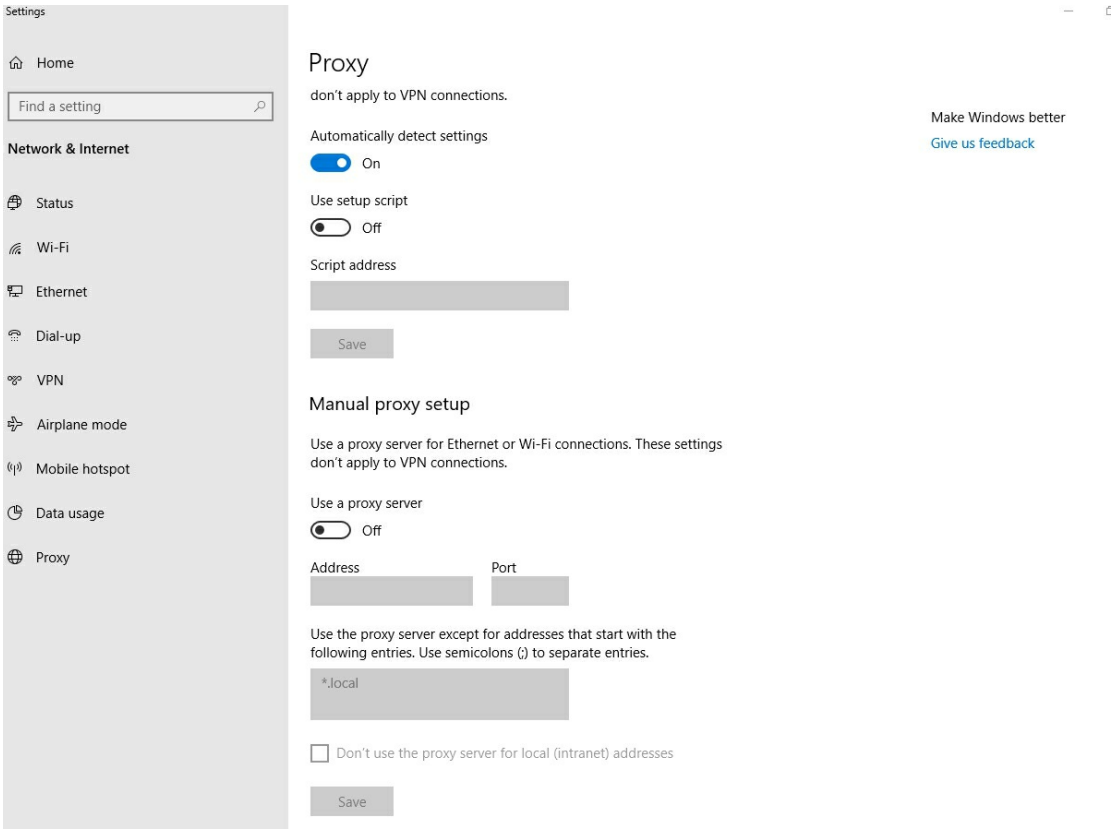
as a proxy server. Proxy servers see connections differently, or rather end users, and the server connections that they're making are all proxied through a single device sitting in the middle of the network. If you were to look at a connection, it would appear to be coming from the IP address of the proxy server rather than the client.

That's the whole idea of a proxy server, is it proxy users connections. If you're using a proxy server, you're going to need to have some specialized coding, and that'll happen on a per application basis. The proxy server generally operates up to layer 7 of that OSI model, and it will intercept connections and recreate connections from itself. Then this gives it the ability to look into the granular details of that application.

The proxy server should be transparent to an end-user, so you may have to set up a proxy server definition in your web browser to define where the proxy server is, or you may have other ways, if you're using a web security appliance, which can act as a proxy server, then there are ways to have it automatically provision the client.

Some protocols can be used, such as Web Cache Control Protocol, or WCCP, that can take traffic and forward it over to a proxy server, but in general, our proxy servers are going to need to be defined by our client. If you were to jump over to a Windows machine and open up Internet Explorer, you could go into the settings, and inside there, you should have options to define the proxy server.

Here's the Proxy setup, you just open up proxy settings, and from here it's telling me to detect proxy settings automatically, but I could say that I don't want to automatically detect it because I want to do a manual proxy set up. If that's the case, I can go ahead and define the address and the port where that proxy server is, and I can start using that as my proxy server. This does take a little bit of additional set up on the client-side to be able to support proxy server technology.



Proxy servers tend to be a little slower than stateful and stateless packet filters, is it's better to get rid of the proxy server, which runs on software on a machine, and combine that proxy server technology with stateful firewall technology, and this way once you have these two things combined, we don't have to worry about other protocols that are on the network other than just web traffic.

Anything that is going through the firewall can now be filtered, and if it's relevant traffic, it can be sent up to the proxy, all the way up to layer 7. This is what the ASA does. The ASA is a proxy server in a way. Cisco IOS routers that are running Zone-Based Firewall features are also proxy servers in a way, but at the same time, they're also a stateful firewall. This leads us into a discussion of what we would call a Next-Generation Firewall or a NGFW, and the reason that I point this out here is because ASAs and Cisco IOS devices that are running Zone-Based Firewall features are Next-Gen Firewalls that include stateless, stateful, and proxy server technologies, and much more. Let's move on and talk about Next-Generation Firewalls.

Chapter 53 Next Generation Firewalls

You will often hear people saying Next Generation Firewalls or Next-Gen Firewalls, but what are they, and why do we need them? Well, it combines a lot of that ability that we've already discussed previously. The stateful firewall capability, the proxy server capability, but then it adds several other features. So first of all, a Next-Generation Firewall would include URL filtering.

URL filtering is going to look for my web requests, and then make decisions based on some kind of a list, maybe it's a white list, perhaps it's a blacklist, it's perhaps a list based on the reputation of websites or something that Cisco devices can do. That's just one feature of a Next-Generation Firewall.

Another feature of these firewalls would be something called the AVC, or the Application Visibility and Control. This feature is going to give me the ability to look up the OSI protocol stack into layer 7. At layer 7, I'm looking at application information, and from there, I can go ahead and control the actions that happen at layer 7. An example of one of these actions would be a “get” or a “put”, which would be functionalities of the HTTP protocol.

It gives me that layer 7 all the way up the protocol stack visibility. I don't have to do this for every single application, but I want to do this for the more common ones, and web traffic would be one of those. Likewise, Next-Generation Firewalls would be what we call context aware devices. Context aware means that the firewall itself understands where things are happening.

It has a big picture of what's going on in the network. It knows who is connected or what kind of device they're using, whether it's an iPhone, an Android device, a Windows laptop, or a Mac. It knows what these different devices are. It also knows where they're connected in the network. For example, are they connected to a network switch, or are they on our guest wireless? It knows the times that they've connected, that's the when, and it knows the how, which is wither wireless, wired or a VPN connection.

From there, using the context, we can have a policy that says for example, if a specific user at a particular time with a particular device tries to connect at a particular location. We're going to allow or deny it. Therefore we've got a lot of control there. Next-Gen Firewalls are also going to include some type of element of intrusion prevention. Intrusion prevention is going to do deep

packet inspection. It's going to look for signatures inside the packet, and these actions are going to be based on rules that are downloaded and kept up to date by the vendor such as Cisco. We can't create our own custom IPS rules, but it's a better idea to use the rules that, for example, Cisco provides by default, and then add our own on a one-off basis when we have what we would call a “zero-day attack”.

Intrusion prevention gives us a deep packet inspection with intrusion prevention rules. Next-Gen Firewalls might also use something called Advanced Malware Protection. We know that Malware is one of the most dangerous things to fight with, and they are constantly transforming, and using different techniques to get around our security controls. But with Advanced Malware Protection, our Next-Gen Firewalls can keep us protected against threats that we see today, and analyze files and compare them against what other people have seen.

That's part of Cisco's strategy, is that they will create a hash of a file, or our firewall would take a file that's being transferred, it would create a hash of it. That hash would be sent to Cisco, not the file initially, and then it'll compare it to see if that hash has been seen by other customers or by Cisco themselves, and if it's Malware, it tells us that's a Malware, and it blocks it.

Usually, it's about 15-20 minutes that it takes for this to happen, which is relatively quick. By the time that happens, we can block this packet in transit. That's Advanced Malware Protection. Lastly, Next-Generation Firewalls can support Identity-Based Access. Identity-Based Access is going to look at who a user is. Then in our access control rules where we would generally write permit TCP from one host to another destination, we can take that a step further and say that we want to permit traffic from the marketing group on the marketing subnet to a specific destination, or vice versa, and deny traffic from the marketing group.

You can also integrate Next-Generation Firewalls with Active Directory. There are a few of different elements that would also be involved. We would have CDA or Cisco Discovery Agent, which is a virtual appliance that runs and communicates using WMI back to Active Directory, and it gets your log in events. Then CDA reports back to your ASA about who is authenticated, so now the ASA can also talk to Active Directory, and query for users and groups. As traffic passes through the firewall, we have the ability now to look

at who is logged in, and have a mapping between their IP address and a log in the event for the user “Jack” in the Marketing group, for example.

We can that Jack is in the Marketing group, so apply an access rule that says permit Jack access over to our server, and I can implement that rule based on the identity of the user. These are only some of the great features that we have in our Next-Gen Firewalls.

Chapter 54 High Availability & Failover

Using a Cisco ASA, we have multiple options for fault tolerance, and to achieve different performance requirements. We have high availability, for example, of active/standby failover, or active/active failover, and we also can configure clustering, not to mention that we can run redundant interfaces. We also support EtherChannel often referenced as PortChannel. Let's First discuss the active/standby failover.

Active/standby failover is when we have two ASAs that are operating in a failover pair. We have ASA1 and ASA2, and both are going to synchronize their configurations with the primary device that is operating in an active mode. When we build our standby configuration, we're going to define one of these devices as being the primary, and the other device as the secondary.

Then the primary should be active, and the second will be standby. The configuration that we have on ASA1 is going to be copied over to ASA2 through something called configuration replication. Then they're going to have an identical setup. None of the traffic is going to pass through the standby device unless there's a failure. If the ASA1 were to fail, then the secondary device would transition to the active state and would now start forwarding traffic for the network.

The configuration matches, therefore you would maintain your access control and the rest of the configurations on ASA2. Well, some of the connections are going to be re-established. For example, if you're doing AAA authentication, you're going to have to re-authenticate connections through the device once the switchover has taken place. So that's active/standby failover. Some devices only support active/standby failover on lower-end devices, but some machines also support active/active failover.

Active/Active failover is a feature that you want to have. Imagine the same scenario as before. You're going to have two ASA devices, and those ASA devices will be operating and forwarding packets at the same time. We have ASA1, and we have ASA2. These devices are going to operate in multiple context mode. In multiple context mode, what we end up with is more than one virtual firewall running at the same time. Imagine that we have context A, and we have that configured on ASA1 and ASA2. Then we have context B, and context B is operating on both ASA1 and ASA2. We have a traffic

flow for context A, and because we're configured for failover, we've got failover replication happening, and the configuration from context A on ASA1 is acting as the primary active device. It copies its configuration over, replicates it to context A on ASA2, which is acting as the secondary device, and it's in standby mode.

At the same time, we've got context B on ASA2 that has a traffic flow through it. It's operating as the primary active device, and it is replicating its configuration over to context B on ASA1, which is acting as the secondary standby device. These would be two different independent firewalls processing traffic differently, but it allows us to do some essential load balancing.

You might imagine that we have a router on either side of a physical ASA, and router1 has context A on ASA1 that provides one route to get across, and then we have ASA2 with context B that provides a secondary route. We can load balance traffic, we can send some traffic through context A on ASA1. In contrast, we send other flows of traffic through ASA2 on context B, and if for some reason ASA1 were to fail, then context A over on ASA2 would then become primary active, or it would become secondary active. Those traffic flows that were being sent up and across, are now going to be sent through ASA2, and it'll take all of the traffic, but it offers us high availability. That's just a brief look at how active/active failover would function.

Chapter 55 Clustering

Another high-availability feature that we want to discuss is known as clustering. Imagine a similar scenario as earlier where we have two ASA devices, but in this case we are going to have 4 ASAs. To be realistic, we could have like 10 or 20 ASAs in a cluster, but let’s discuss if we were to have only 4 ASAs.

These 4 ASAs are going to be connected to a network switch, and this cluster acts as it’s only a single ASA. These ASAs are going to communicate with one another over what's called a CCL or Cluster Control Link, so they'll be able to have synchronized configurations. The links between the network switch and the ASAs can operate as a multi-chassis EtherChannel, which is only one great link that we might choose to configure as a trunk, so it appears that the switch has a high bandwidth connection to a single ASA.

Traffic flows through that ASA, and these devices called ASA1, ASA 2, ASA 3, and ASA4, they will all deal with the flows of traffic, and how ownership is handled, and they'll redirect traffic to the appropriate ASA so that it can handle the stateful traffic inspection.

But, clustering offers me this ability, so if one of the devices in the cluster goes down, then the other devices will still be there, so we have high availability, but we also have some scalability too, because the more prominent links, the more connections per second can be handled. Therefore it’s an advantageous feature on Cisco ASAs.

Chapter 56 Zone-based Firewalls

A Zone-Based Firewall is a concept that we use in Cisco IOS that allows us to group physical and virtual interfaces into a zone, and then apply firewall policy to the traffic that's moving between those zones. This is very beneficial because it makes it easy for us to build in our firewall policies as we add additional interfaces by designating them as a zone. Then our policy will automatically apply to it. It's a modular configuration.

If you think about the Cisco ASA Firewall, it's a purpose-built appliance that's designed for stateful packet inspection, but when we take a router, and we configure its interfaces, and we start passing traffic, there's no state information maintained, even the ASA does it by default, our Cisco routers don't.

Cisco routers used to have a configuration method, or a configuration feature called CBAC. CBAC stands for Context-Based Access Control, and you would configure inspect commands. You would apply those on the interface, either inbound or outbound, and that would cause the router to inspect traffic, which meant that it had to maintain state information for the traffic it was looking at, and that made our Cisco IOS devices stateful.

Well, we don't use CBAC anymore, yet we do see it from time-to-time in the production network, but once we had Zone-Based Policy Firewalls or ZBFWs, introduced, there was no need to do CBAC anymore. We could use a standard configuration method, much like we use for QoS or Quality of Service configuration, class-map, policy-map, or even service policy, and we could apply our firewall policy utilizing that method.

It does provide a stateful packet inspection, and it’s also VRF aware. We use VRFs for isolation, so it's a way to create a virtual router or a virtual route forwarder, which builds its own routing table, and you can link that to interfaces, and isolate your traffic. Zone-Based Firewalls also support URL filtering. The URL filtering is not done the same way that we would do it on the ASA, because we're not talking about integration with a FirePOWER module and using Next-Generation filtering, instead, we're talking about the older method of using filters to take a look at our URLs. Check it against a white list or a blacklist that we've created, or go out externally and look at another type of server that would be doing the URL filtering, and that is

another supported feature of Zone-Based Policy Firewalls.

Likewise, it helps mitigate DOS attacks, and once HTTP inspection came in, many inspections for different protocols, RPC, instant messaging applications such as Yahoo, and MSN Messenger, and several peer-to-peer type protocols could then be filtered. Next, we also have an auth proxy, which gives us the ability to authenticate our users before passing their traffic through the device.

Other features such as stateful failover with our firewalls, IPv6 stateful inspection, support for TCP packets that arrive out of order, and so on. There are more features nowadays, but these are the more important ones that were added since the early days of Zone-Based Firewalls.

Chapter 57 IDS & IPS

IDS, stands for Intrusion Detection Systems, and IPS stands for Intrusion Prevention Systems. To explain the differences between intrusion detection and intrusion prevention, it’s straightforward. Intrusion detection doesn't take action on packets inline. Let's assume that we have a Cisco ASA. That Cisco ASA is going to control access to the network, based on the source IP address and the destination IP address and port numbers.

We do that with an access control list, but an IPS is going to be controlling access, based on the payload inside the packet, and with intrusion prevention, it'll prevent malicious traffic from passing by looking into the payload of the packet, and then determining whether or not that packet should be dropped before we forward it.

That's very different than intrusion detection or IDS. With IDS the way that works is that IDS will generally produce an alert for us whenever it sees offending traffic, something that doesn't match our signature, but it's not responsible for mitigating that threat. In other words, it's typically going to be receiving the packets that it analyzes by some type of a forwarding device.

For example, we might mirror packets over to an IDS, and then the IDS is going to be able to look at them, but by the time the IDS looks at those, those packets already on its way to the destination, so it's an after the fact action. The IDS could help us determine that we need to block packets from here on out, and then it would be able to block subsequent packets, but it's not going to drop the offending packet, because by the time it tries to drop it, it would be too late.

It's an out-of-band type of detection system, so the big difference between the two is that IPS is a prevention device that looks at packets in-line and able to discard them, whereas IDS receives packets that are generally mirrored to the device, and then prevents subsequent packets, but not the initial packet.

To better understand IPS or IDS systems, we should take a look at the history of these devices. About two decades ago in early 2000, Cisco has deployed their first intrusion detection sensors, products that was a dedicated IDS appliance, and as time when by these appliances became capable of handling traffic in-line, and that's when the shift happened from IDS to IPS.

Nowadays, our IPS is a dedicated appliance in most cases, but it can also be

installed as a module on another network device. For example, it can be a module in the ASA, or it could be a module in Cisco routers. Originally two strategies were used in intrusion detection systems. One of those strategies is known as anomaly detection. Anomaly detection simply learns patterns based on what we would consider being regular network activity. For this to happen you need to have a baseline of the network. Until you have a baseline, you don't know what's normal. We let it run, we let it look and watch for some time, and then we define what is being regular network traffic. Then that anomaly detection will just sit there passively, and if it sees something that looks like it deviates from the baseline, then it starts to send us alarms. This is anomalous behavior.

The other type of technology that was used initially, and a type that is still commonly seen is rule-based detection. Rule-based detection is a definition that we call a signature. In this case we define the signature, and say that this is a rule on how traffic should look like, something that we're looking for, that we're going to identify as either being good or bad.

These signatures need to change as the threats change, so we'll use this database of signatures or this database of rules, and it'll use those rules to match against traffic to determine whether something suspicious is going on or not. That's how IDS started, but IPS became very popular, and there were two types of intrusion prevention systems that we would start to see quickly emerging on our networks. Those were either a network-based IPS, and there were several host-based solutions. Network-based solutions are embedded in the infrastructure, and it's just looking at packets as they flow through the network, and it could be comparing them up against the rule database, which are those signatures that we have. With those types of network-based devices, they're not going to be able to see encrypted traffic. They do not have visibility into encrypted traffic strings, and that's because the traffic's already encrypted, and they don't have any knowledge of the encryption keys, so they can't look at it. Nowadays, you instead see encryption proxies, and you'll establish an encrypted session to it, and then it'll establish the encrypted session to your destination.

Everything in the device is clear text, and then it can analyze your data, but traditional network-based IPS doesn't have visibility to that. The host-based IPS that we would see is an agent-based install. This is something that we would put on every single machine. Back in a day, the Cisco Security Agent

or CSA was used by Cisco. It was the Cisco flagship at the time, which was a host-based IPS product for a long time, and that was acquired by a company named Okena.

Okena made the StormWatch and StormFront products, and they would be able to implement agents. They had a management center, so the Okena StormWatch and StormFront, that got rolled into what was called the Cisco Security Agent, and then we had the management center that would be able to manage and help us deploy those agents and configure policies.

The benefit to using those host-based systems is that it can detect intrusions that utilize encrypted communication. Because they see it on the device before the encryption happening, or after it's been decrypted, it gives them more visibility. Therefore these devices are complementary for types of intrusions that don't generate network traffic, or something that would be internal on a machine. They all have their place, but they're not as popular as other network-based intrusion prevention devices would be.

Moving on, there are a few terms that you should be aware of when it comes to intrusion detection and prevention systems. The first term is vulnerability. We already discussed the term vulnerability early on at the beginning of this book. Still, as a refresher, vulnerability is a weakness that a device has that could allow it to be compromised. It is a weakness.

It could be a weakness in software, it could be a weakness in location, it could be a weakness in hardware, but that's what vulnerability is, it's something that compromises the security of the system or even the functionality of the system. That's not to be confused with an exploit. An exploit is what we use to leverage the vulnerability in hopes of compromising the security or the functionality of that device. An exploit might be a RootKit. That might be the tool that we use to leverage a particular vulnerability.

We have a vulnerability and exploit, but there is another term that we need to understand, called threat. A threat is the perceived potential that something can cause you harm. People often talk about threat levels. The government might have different threat levels. As an example, in the US the Homeland Security System is a color-coded terrorism threat advisory scale. If the threat level is low, then there's low risk of attack. If the threat level is elevated, there's a significant risk, and if it is severe, there is severe risk of attack.

We have threats that we perceive to be on a certain level. The higher the level

the more urgent the issue is. Let's imagine that we have a risk that exploits the vulnerability in our Windows web server that we're aware of. We have a threat that exploits a vulnerability in our Windows web server. That could be harmful, and we could consider that to be a different level of danger, and this leads us into the term Risk.

Risk is the likelihood that a particular threat, using a specific attack, is going to exploit a system, or it's going to leverage one of those vulnerabilities. Hence, the level of risk is the threat level. What is the threat level? What is the risk level? Back to our example of having a threat that exploits a vulnerability in our Windows web server, but if you're a Linux web server, then the risk is low that you're going to be vulnerable to that threat.

Yet, if you're a Windows web server, and we're looking at an attack that exploits a vulnerability in Windows web servers, then the risk is much more significant that you are going to be susceptible to the attack. If we were going to relate these together, the concepts of threats, vulnerabilities, risks, and the value of an asset could be linked in that risk equals threat times the vulnerability times the value of the asset. This is useful when we're trying to prioritize things to the business.

In this case, patching Windows servers is more important to us because we need to focus our efforts on handling the highest risks first. Once we start looking at our real environments, and we have these security controls such as our IPS that are trying to determine if something is important to us, well, sometimes even legitimate activity that we see could be marked as malicious by the IPS. Maybe something is malicious, and it's not marked as malicious by the IPS, and there's a reason for that.

The reason for that could be because all security controlled decisions can be classified as either being a true positive, which means that the IPS acted. After all, it saw something malicious, and it did exactly what it was supposed to do, so it was a true positive, or it could be considered a true negative. The IPS doesn’t do anything because there was no malicious activity, so that's normal. Those are two ways that the security controlled decision could be classified, but then we can get into the two that are not very desirable, and they just end up being a consequence of how our IPS devices function.

We have a false positive, which means that our IPS did something. It took an action or it alerted me even if the activity was not malicious. Usually,, this is

because we have controls that are too restrictive, so they're not permitting the legitimate traffic. Or, it could be because we have reactive controls, and the description of what is an attack is just too broad, so everything looks like an attack, even if it's not.

The other type of signature that we have is called false negative. A false negative is perhaps the worst case. The IPS didn't do anything even if there was something malicious, and that is generally because the proactive controls are undisturbed. We permit more than we should, or the reactive controls are too specific, and nothing matches. Because of this, even if we might be able to figure out the risk by looking at the threat and the vulnerability and the value of the asset, and we are be able to match up against the signatures that we're looking at, the decisions that the device makes could fall into one of four categories.

If it falls into the false area, false positive or false negative, then we've got some tuning to do with our devices, and we're going to probably spend a lot of time working on getting these devices acting normal, which means true positive and true negative versus acting incorrectly with false positives and false negatives, so we want to avoid those errors, and it's all part of the tuning of our security devices.

One of the differences between an IDS and an IPS is how they can respond or how they can take action when something is seen. For example, an IDS is passive in its response because it can't do anything to real-time traffic because of the way that it's implemented. It's not necessarily in line with the traffic. It sees a copy of the traffic rather than the actual traffic. Passive actions would include sending an alert, or monitoring a flow of traffic. When we send an alert, this essentially means that we're logging a message. We want to do this when we see something malicious. We want to log when malicious traffic is recognized, and then we want to store this in a database that we can review, and we don't want this traffic to be overwritten.

Often, we'll send these to event viewers that we're going to be monitoring, and this could be happening in real-time, up on large screens in our network operation center, or we could be storing it and have a set time, that we want to go in there. If you're in a large organization, you're going to be doing something like SIEM or Security Information Event Management, and you'll be using products such as Splunk. An alert is going to be stored in those types

of devices.

It's going to be visible for us, so that we can view it, and we can figure out what's going on. Often you'll hear alerts called alarms. They're the same thing. It's a generation of a message when we detected something that was perhaps malicious. That other option is to monitor. When we monitor, that means that we are taking our resources and concentrating them on traffic that's associated with a specific set of hosts.

For example, communication between point A and point B, and as we monitor communication between point A and point B, we are likely going to be capturing that traffic, and this might be raw packet captures that we can open up in Wireshark, and analyze later on. The monitor is more involved than just sending an alert. Our active actions, or responses that we would only see with an IPS device, is a drop.

When we drop packets, it’s because the packet looks bad to us, so it will not being forwarded, and it will be discarded. That prevents packet from reaching the destination. Another option that we could do is send a reset. When I do a drop, the source and the target don't know about anything that happened. They just know that the packet wasn't received, so they may request it again at a higher layer, using TCP. But, when I use a reset if I see some suspicious payload, and my action is set to do a reset.

If this is a TCP connection, then my sensor can inject a TCP reset, and it'll send that in both directions to the source and the destination, resetting the TCP connection, which substantially drops it, clears up all the resources on the host, and the destination device is not waiting for the packet anymore. We can also do a “block”, and this is something that works with TCP, UDP, and ICMP as well.

We are going to block a specific session, and it tracks the source and destination, or just the source that should be blocked, and then if we wanted to, we can clear that block. Either way, we can block a session or we can block a host. Keep in mind that if we block a host, we're blocking all traffic for that particular host, so when we select our actions in IPS, we need to tell it whether it's a block session or a blocked host.

Then finally, we have an option for “shun”. Not all IPS systems are going to have the ability to do this, but “shun” tells my sensor that’s integrated with an ASA, and it can talk to an ASA, it can connect to an ASA, it can SSH into an

ASA, and it can issue a “shun” command for a particular host who should be blocked.

This way, the ASA could be on the other side of the network at the perimeter, and that “shun” could occur somewhere other than where the IPS is located, so this is beneficial for pushing out a “shun” to various parts of the network. We could have that temporary or manually removed. Generally, the SIEM can control the “shun” functionality, but it's just a way to do a block, and it's a way that we do a block remotely. These are some of the terms and the actions that you'll need to be familiar with when it comes to IDS and IPS technologies.

There are a few evasion techniques I just want to mention. Attackers are smart, and they use various evasion techniques. One of those is called a packet fragmentation technique where all you're going to do is use fragmentation to try to bypass or trick the sensor into thinking that your traffic is not bad. One way to do it is to change the fragment offset values, so when the packet is reassembled, it's reassembled differently than it usually would.

Once you think about it, we have a device, such as a router, and it decides that a packet's too big, so it splits it up into little fragments, and each fragment has an IP header that looks the same, but the flag inside and the offset values are going to be different. The flag bits and the offset values are what the host at the end is going to look at, and then reassemble the string.

The device will going to look at the first packet and it's offset, so, for example, a fragment offset value is 0, and it knows that it’s the very first packet, and then there's a flag, and that flag determines whether there are more fragments or not. If I receive a string of packets, and the more fragment bit is turned off, then I know that's the last fragment in the string.

If I'm the end host, all I have to do is to use the fragment offset values to rebuild that packet. Primarily what happens with a packet fragmentation evasion technique is that the attacker hides their data in a string of fragments, and you can use applications such as Fragroute to do this. You can craft it yourself, and it makes it an easy way to sneak traffic back if your network's not watching for this type of fragments.

Another attack is an injection attack. In the case of an injection attack, the attacker is going to put certain strings in the attack code, so that you can't tell

that it's an attack code. If we look at the overall attack code, we have to pull those strings that don't belong there to see that it's an attack. I can fragment the packet, and then I can inject the traffic, and one of the evasion techniques that we have to help us with this is called “string reassembly”.

Our sensors are typically able to do this. Then there are the alternate string expressions evasion techniques, and that's where you're using different types of strings of traffic that are different encodings but might mean the same thing. Perhaps if we don't look for a kind of encoding, and we allow them to sneak traffic through, using that type of encoding, well now they're in the network, and we're in trouble. As I mentioned earlier, there are many evasion techniques, but both IPS and IDS devices are helpful devices even if you let them run with their default configurations.

Chapter 58 Security Intelligence Blacklisting

Before any attack happens, we can receive some information from Cisco that's based on their latest reputation intelligence. Cisco provides us this intelligence feed, and based on the information in that feed, and we can configure a global blacklist. We can work with that global blacklist, and we can import third party information, but this is our very first line of defense. This is what we would be doing before an attack happens. We would be retrieving the information from Cisco, and we would be implementing a default rule set that is going to blacklist those who have a poor reputation. Blacklisting is an important aspect, but blacklisting is not the only line of defense that we have beforehand. We also have what's known as AMP or Advanced Malware Protection. With Advanced Malware Protection, we can protect the endpoint, and we can also protect the network, so we have two components, endpoint, and network. The endpoint product, Cisco AMP for endpoints has a connector that you install on an endpoint. Then it uses a cloud-based service that takes the malware detection, and offloads that burden out to Cisco to the cloud-based service, and that way it helps us to manage malware, quarantine it, and to protect various endpoints. We also have what's known as AMP for Networks. This uses our FirePOWER based appliances, and this will help us to detect malware that's in transit, so we can look at the network connections we identified in transit. It also talks to the cloud, and it gets very current information that comes back with something called a file disposition. The file disposition comes back and gives us information on whether this is known malware or not. Then with the network-based AMP, our system will inspect the network traffic, and it looks for several different types of files, and then it can store detected files, and do folder analysis. We can submit that traffic up to our Cisco Collective Security Intelligence Cloud, and then it'll do a lookup based on a hash value, or it can do a more detailed look by running this in a sandbox. This is something that was using the Advanced Malware Protection capability so that we can identify traffic ahead of time, and this helps to protect us before the attack.

Chapter 59 Email Security

We all know that email is a critical tool in a business environment. Whether

we use it for business communication or personal communication. It's just the way that people talk, and even if we've got all these social networks that are available now in different messaging apps, people still revert back to email as their default form of messaging and communication. Knowing that, and how many users are accessing email daily, it exposes us to several threats.

What kind of threats are we talking about? Well, we all know about spam. Spam's bad because it is a time-waster. First, we start getting unsolicited emails. You open your mailbox, and you have all this junk mail in there, and that's what spam is. But today's threats are even worse. We've got spam that contains viruses and malware.

Often that can be used to get you to click on something, so it comes in the form of a spam message. It wants you to open up a file, maybe it says that your shipping request didn't work, and they sent you a transcript, and you need to open this zip file to read it, and you do, and all of a sudden you're now hit with malware.

It can also include fraud and impersonation. Somebody can impersonate someone else. It's not something that should be taken lightly, and it's easy to impersonate emails. You can send an email that's got a formatted HTML page. We have these links that look like they're real, but they're not. They'll try to get you to enter password credentials to your bank or reset a password, and all of a sudden, you've got your identity stolen or your login information for your bank stolen.

It could also be impersonating someone that you think you know telling you that they're in trouble, and they need you to Western Union them some money. I've seen those emails come across, so there are all kinds of bad emails that come across, so we need a protection mechanism for email that's more than just a junk mail filter.We need something more than just filtering that email. We need something that's looking at it, and determining if the content is malicious, even though it might look like it's coming from a legitimate source, so that's where the ESA or the Email Security Appliance is coming in. In general, the first and last server where we see an email coming in, or going out from, or to the internet, it's going to be the email server. We see the path out of our network towards the internet, and it's going to be filtering spam and viruses, and malware, and protecting against all of those different types of threats. How does it do this? Well, let's just attack spam

first of all. There are two options that we can use to filter spam email messages.

We have reputation-based filtering and we have context-based filtering. Beginning with reputation-based filters, essentially, what we need to do is figure out if a server is known for sending spam. That's a little bit too big of a task for us to be able to do, but not with Cisco's help. With Cisco's help, we can use what's known as “Sender Base”.

Sender Base is the largest repository for security data, and it's got all kinds of information there, such as sources of spam, botnets or malicious hosts. We can download this data from Cisco's Sender Base, and then we're going to use that, as our first line of defense.

Email's going to come in, and we're going to look at the source IP address of the email server. We want to know which email server sent it, and we're going to compare that against the Sender Base data, to figure out if the sender is known for sending spam. The data that's in Sender Base has a score that's attached to it. It's a composite score, and the values are from -10 to +10.

There are some default actions that we can take. We can modify these, but here's how it works. Any traffic that comes in that is between +10 to a -1, and we're going to accept it. If they have a good reputation score; go ahead and accept those emails. If mail comes in that is between -1 to -3 we are going to accept it, and we're going to throttle any emails that are coming from that same server.

If they are trying to send us a whole bunch of spam messages, they're going to be throttled because they're questionable in their score. Finally, if they're reputation is anywhere between -3 to -10 that's the low end of the scale, well then we're just going to block those data, and we're going to drop those emails and they're not going to get those emails past us. That's reputation-based filtering, and it does require that we have Sender Base in use, and we're able to talk to Cisco and get that data.

The other option that we have is called context-based filtering. Context-based filtering uses information that it obtains from inspecting the entire message. It's going to look at the attachment, it's going to look at the identity of the sender, it's going to look at the content, it's going to look at any URLs that are in the message, any formatting of the email, and it's going to use all of this information. It's going to run it through some algorithms, and determine

whether or not this is a spam or not, and the idea is that we don't want to block anything that's legitimate traffic.

Legitimate emails should not be blocked, so we should use reputation-based filtering as that’s the first line of defense where there's not a lot of processing done on our part. We're just comparing to a list that Cisco's already made for us, and they've got that largest repository of data at Sender Base, so we can trust that data, versus context-based, which is more like a second line of defense, and that's how we should filter spam.

What about viruses and malware? Well, in terms of viruses and malware, the ESA uses a simple multi-layer approach. First of all, it uses the first layer of defense, called outbreak filters. What's that? Well, an outbreak filter is, something that is downloaded from Cisco's Sender Base. This is a list of known bad mail servers, and they get generated by looking at global traffic patterns.

Everything that Cisco can see, and they look for anomalies that are associated with an outbreak, and if there is an outbreak, we get this data quickly, and we can filter based on it. The second layer of defense is an antivirus signature, and primarily, we scan quarantined emails, making sure that they don't have any viruses, and then we go from there. This is considering inbound email. What about outbound email? Well, ESA does scan outbound email as well, and that's designed also to provide us with outbreak protection, antivirus protection, making sure that nothing wrong is being sent out of our network.

Chapter 60 Data Loss Prevention

When it comes to Data Loss Prevention, we're talking about making sure that personal data doesn't leave our organization. These data include people's identification, credit card information, social security numbers, intellectual property, or any data that makes money to an organization. These are all things that we want to prevent from getting out, and Cisco ESA does support Data Loss Prevention. It's got a feature in a software that does this, and Cisco partners with RSA, who also has a DLP solution already doing this.

In this partnership, they can provide integrated DLP right on the ESA, and this is a licensed feature, so it doesn't just come automatically, but if you wanted to prevent data loss, we can go ahead and add it as an extra feature. Then there is another feature of our Email Security Appliance called Advanced Malware Protection or AMP.

AMP has three aspects. It's got file reputation, file sandboxing, and retrospective file analysis that can all be used to identify and stop threats. As far as file reputation goes, every time we see a file come through, we grab a fingerprint of that, which is a hash. That hash is then sent to Cisco's cloud-based intelligence network, and it comes back with a verdict. Is it reputable or not? Is it good or bad?

If it's something that Cisco doesn’t know whether it's good or bad, then we can analyze that file by putting it in a sandbox, running it in a secure location. Then the disposition will come back from that sandboxing, which will be then fed back into the file reputation database, and then everybody gets updated on that information. Then we also have file retrospection, which is all about going back and looking at traffic that is something that we don't know right now, but it might be later deemed to be a threat.

The reason that this capability is provided is that malware's gotten very sophisticated. Different techniques can be used, such as obfuscation, sleep timers, polymorphism, and the idea is that they don't execute themselves until they've been in your network for a while. They sit there and they hide, and then after a while, they start executing themselves. The retrospective file analysis is something that solves that problem of files passing through, and then, later on, prove to be malicious, so we can go back and mark them as malicious, and then we can see where they came in. These are just some of

the features of the ESA. We already discussed SPAM filtering, virus and malware filtering, DLP, and AMP.

There are several benefits that we gain from using an ESA. We're able to protect against the entire attack range, and we can control email, we can prevent data loss, we can do rate limiting. We have excellent monitoring capability, and the ASA also supports high availability. With that said, let's talk about the message flow of a standard email message.

Imagine that we have a user sitting out on the internet and that user wants to send us an email, so he forwards that email over to his email server, and that mail server is going to do a DNS lookup, and try to find out the information for the destination email address on the address that we have sent it to.

He comes back with a response, and he says, it's this email server. That email is routed through the internet, and sent to wherever that destination record sends it. On our inside as that message comes in, it comes in through our ASA. From the ASA, it goes over to the DMZ where our ESA sits. The ESA then filters that traffic, and forwards it in to our internal mail server, and then our internal mail server delivers that to the email application, such as Microsoft Outlook.

That's the whole process, but what about in the reverse direction? Well, in the reverse direction we have an mail client who forwards traffic to the mail server internally. That mail server then forwards it to the ESA. The ESA then filters, and once it's filtered, it does a DNS lookup trying to find the owner of that record.

After it finds the owner of that record, it forwards it back through our ASA out to the internet, and then to that destination mail server. That destination mail server then can deliver it to the client on the remote end, and that's our standard mail flow. The one catch here is that when the ESA filters that traffic it could be to the point where it denies that mail from being sent, based on our DLP policies, but that is our outbound message flow.

Data security enforcement is designed to give us visibility into areas where data might be transferred out of my network, such as FTP, so this would give us visibility to look at outbound FTP traffic, as well as HTTP and HTTPS traffic. We can use this to prevent the sending of certain files by email or uploading them to external servers. There is some native FTP protection that gives us total visibility into FTP, and it can help enforce our acceptable use

policy, enforce our data security policies, and also can help prevent malware. There is something called the Dynamic Vectoring and Streaming engine or DVS, and that looks for downloaded content, so it'll help us to scan content and prevent for malware and spyware and that's part of the data security enforcement capabilities.

The last feature that I want to highlight is the management and reporting capability. We can look at the information on various threats, and we can generate anti-malware reports, web reputation reports or websites reports that we can look at. This is also to give us visibility into all of these different areas so we can get a complete picture of what's going on in the network in terms of malware defense, data security or acceptable use policy, and how it's being handled.

It also does support SNMP, and it has an alert engine that's built into it that will let us know if there's any hardware problem, if there's any security issue, performance problem or availability problem. There is an integration with LDAP, as well as Active Directory if we wanted to do some authentication, and we can tie that into our access control policy, so that falls under the management realm, and there are some extensive logging that will help us keep track of what's going on.

It also supports some of the standard log formats such as Apache or Squid, and we can also do some custom formats if we wanted to. You can turn individual logs on or off, you can set them to roll over when they've hit a specific size, and it’s flexible in its deployment, so it’s the product that you should have if you've got a lot of web traffic that you're concerned about.

Conclusion

I hope this book was able to get you started on your pursuit of becoming a Cybersecurity Specialist. In case you found some of the techniques and strategies being advanced, no worries, because on-going practice will help you to become an IT Professional in no time.

Thanks again for purchasing this book.

Lastly, if you enjoyed the content, please take some time to share your thoughts and post a review. It'd be highly appreciated!

About the Author

Hugo, originally from Austria, currently living in the Manchester, UK. Hugo is an IT Security Specialist, having over 17 years of experience within the IT field.

He started working on Service Desk, and then moved onto the field of Networking, where partaken various projects including Wireless Deployments, Wireless Security Design, Wired Network Security and Firewall Security.

In 2015, due to the rise of Cyber-attacks, the Security Department was expanding, and began recruiting additional members of the team. This is when Hugo once again made a switch, and started working as an IT Security Analyst.

Since 2017, Hugo become a Security Specialist and began providing professional services and consulting various Companies to improve their security.